4. Compute

Amazon EC2

O Amazon Elastic Compute Cloud (Amazon EC2) é um serviço Web que disponibiliza capacidade computacional segura e redimensionável na nuvem.

O Amazon EC2 oferece a maior e mais abrangente plataforma de computação com a possibilidade de escolha de processador, armazenamento, rede, sistema operacional e modelo de compra.

Planos

Nível Gratuito: O nível gratuito da AWS inclui 750 horas de instâncias Linux e Windows t2.micro (t3.micro para as regiões nas quais t2.micro não está disponível) todo mês durante um ano. Para permanecer no nível gratuito, use somente microinstâncias do EC2.

Há 3 tipos de planos para se contratar os serviços EC2, sendo elas:

- On Demand: servidor virtual, cobrado por second
- Reserved: virtual, por contrato, (medido em USD/hora) por um período de 1 ou 3 anos
- Dedicated Hosts: servidor físico, é possível utilizar licenças existentes dos sistemas operacionais, pode ser cobrado por hora ou por reserva
- Spot Instance: são instâncias alocadas quando existe oferta para uma faixa de valor que o usuário que pagar

Nome das Instancias

- C = Compute Optimized (focadas em CPU)
- R = Memory Optimized (feito para aplicativos que utilizam muita memória)
- G = Graphics (focado processamento gráfico)
- D = Dense Storage (focado para servidores de arquivos, por exemplo)
- M = General (uso geral)
- I = High Speed Storage (focado em banco de dados)
- F = Programação
- T = WebServers
- P = GPU (minerar bitcoin / machine learning)
- X = Memória (servidor apache, por exemplo)

Security Groups

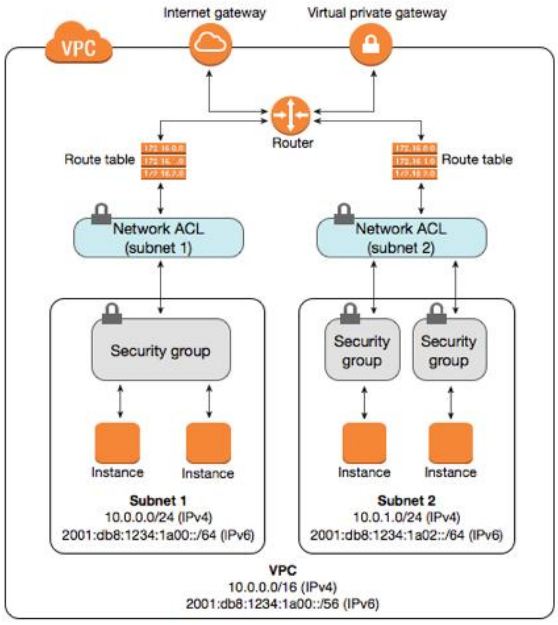
A regra básica é, todas as portas de acesso são Dany (bloqueadas), é necessário permitir somente o necessário. Os Security Groups funcionam por região e toda região por default tem um Security Group associado.

É possível determinar o tipo de trafego de entrada em Inbound, e determinar a saída em Outbound. Por default Security Groups são Statefull, ou seja, sempre haverá uma regra de saída para tráfego de entrada.

As regras quando alteradas e salvas, são aplicadas de imediato. É possível também aplicar mais de um Security Group por instância

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign five security groups to the instance. Security groups act at the instance level, not the subnet level. Therefore, each instance in a subnet in your VPC could be assigned to a different set of security groups. If you don't specify a particular group at launch time, the instance is automatically assigned to the default security group for the VPC.

AWS Security groups are stateful. It means that you do not need to open the outbound for responses - open only inbound for requests. If you think your instances will be sending requests to certain IPs (for example: to upgrade/install a package), then you need to open the IP/port for that request. By default, it is open for all traffic.



Políticas IAM

É possível anexar políticas IAM a uma instância EC2, ou seja, não é necessário adicionar uma Access Key e Secret key dentro da EC2 pois, aplicando uma política a instância, será possível ter acesso somente aos serviços que fazem parte da política sem ter uma chave de acesso cadastrado dentro da instância. Dessa forma, elevando a segurança para caso a máquina seja acessada indevidamente o atacante não terá total acesso programático aos serviços

Volumes EC2

Dentro do EBS - Elastic Block Storage (virtual disk) existem volumes que permitem que se rode um sistema operacional e existem volumes que não permitem.

Utiliza também como unidade de medida para saber a eficácia dos discos a sigla IOPS (Input/Output Operations Per Second), HDs SSD fazem de 5000 a 100.000 IOPS.

Os tipos de volumes são:

- GP2 – utiliza discos SSD, mais caro $, de 3 a 10.000 IOPS
- IO1 (Provisioned SSD) – Alta intensidade (voltado para banco de dados), mais de 10.000 a 20.000 IOPS
- ST1 – utiliza HDD, utilizado para dados e logs. Não Boot (não roda um S.O), não pode ser (C:)
- SC1 (Cold HDD) – dados e logs e é mais barato que o ST1. Infrequent Access, não pode ser (C:)
- Magnetic (Standard) – HDD, Infrequent Access, aceita Boot (pode ser C:)

EBS vs Instance Store

EBS: O Load dos dados é rápido, é possível adicionar mais volumes, os dados ficam salvos no volume, é possível dar um Start/Stop e Terminate

Instance Store: É um armazenamento local onde os dados ficam armazenados temporariamente, quando ocorre o Restart/Terminate da instância os dados são apagados. Esse tipo de armazenamento era mais comumente usado no início quando o serviço EC2 foi lançado tendo grande possibilidade de no futuro ser removido da AWS

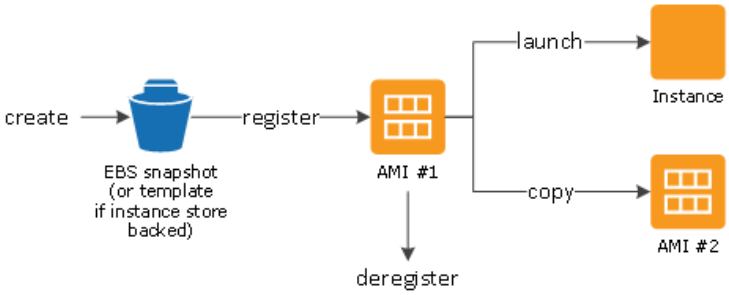
Imagem de Máquina da Amazon (AMI)

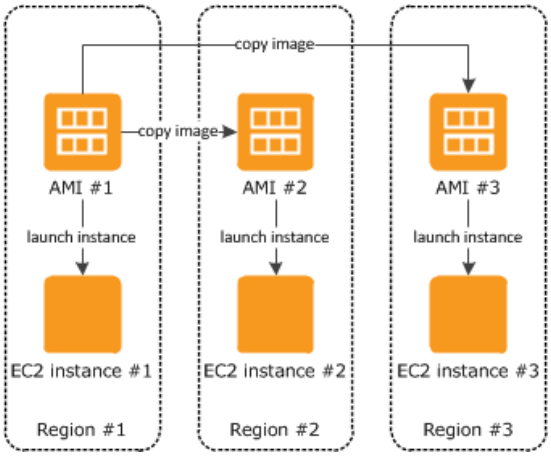
Uma Imagem de máquina da Amazon (AMI) fornece as informações necessárias para iniciar uma instância. Você deve especificar uma AMI ao iniciar uma instância.

Você pode executar várias instâncias em uma única AMI quando precisa de várias instâncias com a mesma configuração. Você pode usar AMIs diferentes para executar instâncias quando precisa de instâncias com configurações diferentes.

Uma AMI inclui o seguinte:

- Um ou mais snapshots do Amazon Elastic Block Store (Amazon EBS) ou, para AMIs com suporte de armazenamento de instâncias, um modelo para o volume raiz da instância (por exemplo, um sistema operacional, um servidor da aplicação e aplicações).
- Permissões de execução que controlam quais contas da AWS podem usar a AMI para executar instâncias.
- Um mapeamento de dispositivos de blocos que especifica os volumes a serem anexados à instância quando ela for executada.





Copiar um AMI

Você pode copiar uma imagem de máquina da Amazon (AMI) em ou para regiões da AWS usando o AWS Management Console, a AWS Command Line Interface ou SDKs, ou a API do Amazon EC2, sendo que todos oferecem suporte à ação *CopyImage*. Você pode copiar as AMIs baseadas no Amazon EBS e as AMIs com armazenamento de instâncias. Você pode copiar AMIs com snapshots criptografados e também alterar o status de criptografia durante o processo de cópia.

Copiar uma AMI de origem resulta em uma AMI de destino idêntica, mas com seu próprio identificador exclusivo. Você pode alterar ou cancelar o registro da AMI de origem sem afetar a AMI de destino. O inverso também é verdadeiro.

Utilizando Meta-Data

Dentro de todas as instâncias EC2 é instalado uma ferramenta para ter acesso a metadados relacionados a instância como: Public, Private, Hostname, Mac, Inst. ID, Sec. Groups

Para utilizar esta opção é necessário acessar via terminal utilizando curl o seguinte comando:

$ curl http://169.254.169.254/latest/meta-data/

$ curl http://169.254.169.254/latest/meta-data/hostname

revoke-security-group-ingress

Remove as regras de entrada (entrada) especificadas de um grupo de segurança.

Server refused our key

Why am I getting a "Server refused our key" error when I try to connect to my EC2 instance using SSH?

I'm receiving the "Server refused our key" error when connecting to my Amazon Elastic Compute Cloud (Amazon EC2) instance using SSH. How can I fix this?

There are multiple reasons why an SSH server (sshd) refuses a private SSH key. The following are some common reasons you might receive this error:

- You're using the incorrect user name for your AMI when connecting to your EC2 instance. The usual user names are ec2-user, ubuntu, centos, root, or admin.
- The user trying to access the instance was deleted from the server or the account was locked.
- There are permissions issues on the instance or you're missing a directory.
- You're using the incorrect private key file when connecting to your EC2 instance.
- SSH server settings in /etc/ssh/sshd_config were changed.
- The operating system couldn't mount (/etc/fstab) home directories.
- You're using an SSH private key but the corresponding public key is not in the authorized_keys file.
- You don't have permissions for your authorized_keys file.
- You don't have permissions for the .ssh folder.

What is Amazon bastion?

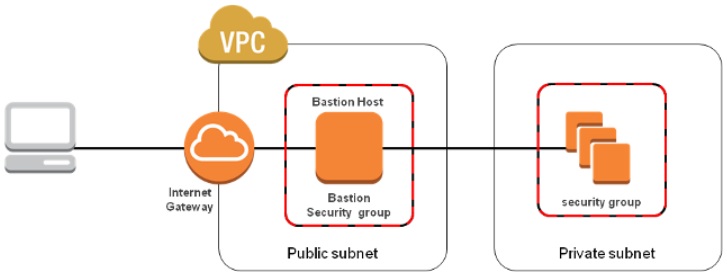
A bastion is a special purpose server instance that is designed to be the primary access point from the Internet and acts as a proxy to your other EC2 instances. ... To define the source IPs that are allowed to connect to your EC2 instances' RDP port (TCP/3389), you configure the instance's security group rules.

Bastion hosts are there to provide a point of entry into a network containing private network instances. ... When using a bastion host, you log into the bastion host first, and then into your target private instance. Because of this two-step login, the bastion hosts are sometimes called "jump servers."

Um host bastião é um computador para fins especiais em uma rede projetada e configurada especificamente para resistir a ataques.

Os hosts bastion oferecem acesso seguro a instâncias do Linux localizadas em sub-redes privadas e públicas de uma Virtual Private Cloud (VPC)

A bastion host is a server whose purpose is to provide access to a private network from an external network, such as the Internet. It does not act as a proxy to route traffic from the internet to private EC2 instance.



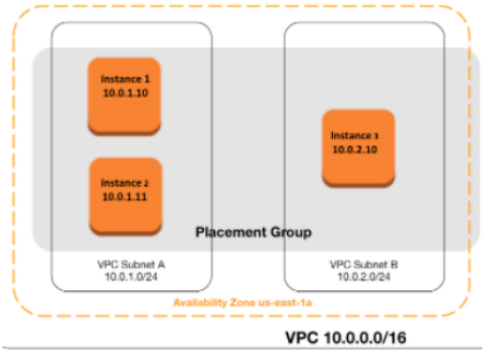
Placement groups

Amazon Web Services' solution to reduce latency between instances involves the use of placement groups. As the name implies, a placement group is just that -- a group. AWS instances that exist within a common availability zone can be grouped into a placement group. Group members can communicate with one another in a way that provides low latency and high throughput.

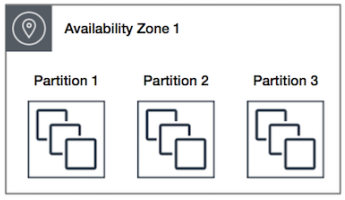
Cluster Placement groups are recommended for applications that benefit from low network latency, high network throughput, or both. The majority of the network traffic is between the instances in the group. To provide the lowest latency and the highest packet-per-second network performance for your placement group, choose an instance type that supports enhanced networking.

Quando você inicia uma nova instância EC2, o serviço EC2 tenta colocar a instância de tal forma que todas as suas instâncias sejam espalhadas pelo hardware subjacente para minimizar as falhas correlacionadas. Você pode usar grupos de posicionamento para influenciar o posicionamento de um grupo de instâncias interdependentes para atender às necessidades de sua carga de trabalho. Dependendo do tipo de carga de trabalho, você pode criar um grupo de posicionamento usando uma das seguintes estratégias de posicionamento:

Cluster – pacotes de instâncias próximos dentro de uma zona de disponibilidade. Essa estratégia permite que as cargas de trabalho alcancem o desempenho de rede de baixa latência necessário para a comunicação entre nós fortemente acoplada, típica de aplicativos HPC.



Partition – espalha suas instâncias entre partições lógicas de forma que grupos de instâncias em uma partição não compartilhem o hardware subjacente com grupos de instâncias em partições diferentes. Essa estratégia é normalmente usada por grandes cargas de trabalho distribuídas e replicadas, como Hadoop, Cassandra e Kafka.

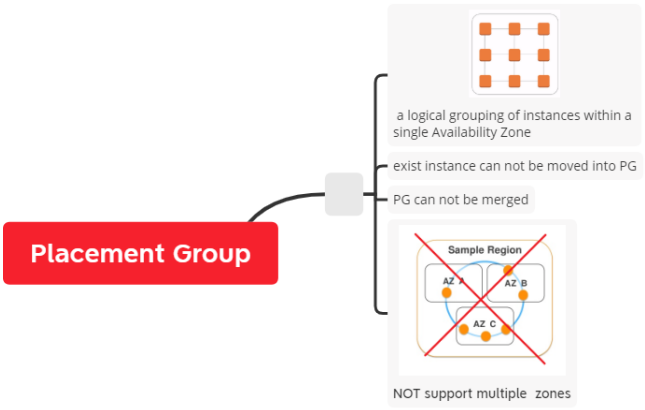


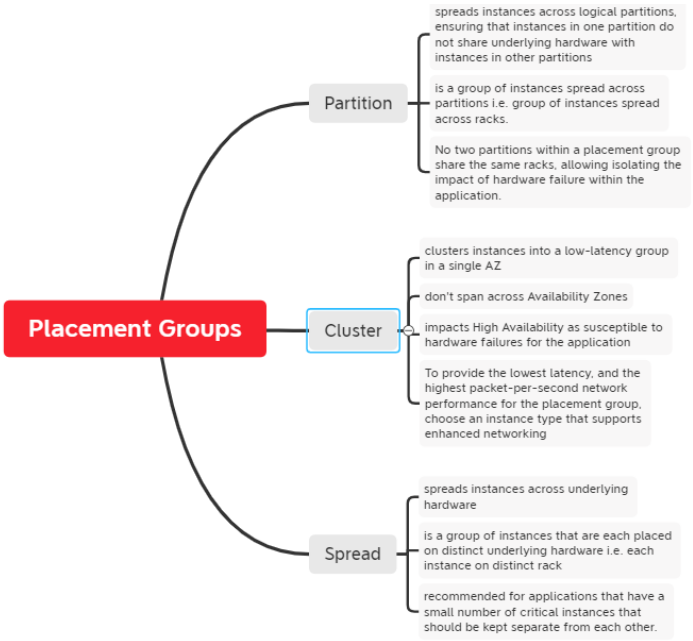
Spread – coloca estritamente um pequeno grupo de instâncias em hardware subjacente distinto para reduzir as falhas correlacionadas.

Não há cobrança para criar um grupo de canais.

Características

You can migrate an instance from one placement group to another **but cannot merge** placement groups.





ENI - Elastic network interfaces

An *elastic network interface* is a logical networking component in a VPC that represents a virtual network card. It can include the following attributes:

- A primary private IPv4 address from the IPv4 address range of your VPC
- One or more secondary private IPv4 addresses from the IPv4 address range of your VPC
- One Elastic IP address (IPv4) per private IPv4 address
- One public IPv4 address
- One or more IPv6 addresses
- One or more security groups

- A MAC address
- A source/destination check flag
- A description

You can create and configure network interfaces and attach them to instances in the same Availability Zone. Your account might also have *requester-managed* network interfaces, which are created and managed by AWS services to enable you to use other resources and services. You cannot manage these network interfaces yourself. For more information, see Requester-managed network interfaces.

This AWS resource is referred to as a *network interface* in the AWS Management Console and the Amazon EC2 API. Therefore, we use "network interface" in this documentation instead of "elastic network interface". The term "network interface" in this documentation always means "elastic network interface".

Bring your own IP addresses (BYOIP) in Amazon EC2

You can bring part or all of your publicly routable IPv4 or IPv6 address range from your on-premises network to your AWS account. You continue to own the address range, but AWS advertises it on the internet by default. After you bring the address range to AWS, it appears in your AWS account as an address pool. BYOIP is not available in all Regions and for all resources.

User data and Shell Scripts

If you are familiar with shell scripting, this is the easiest and most complete way to send instructions to an instance at launch. Adding these tasks at boot time adds to the amount of time it takes to boot the instance. You should allow a few minutes of extra time for the tasks to complete before you test that the user script has finished successfully.

When you launch an instance in Amazon EC2, you have the option of passing user data to the instance that can be used to perform common automated configuration tasks and even run scripts after the instance starts.

You can pass two types of user data to Amazon EC2: shell scripts and cloud-init directives. You can also pass this data into the launch wizard as plain text, as a file (this is useful for launching instances using the command line tools) or as base64-encoded text (for API calls).

Enhanced networking on Linux (Elastic Network Adapter - ENA)

A rede aprimorada usa virtualização de E / S de raiz única (SR-IOV) para fornecer recursos de rede de alto desempenho nos tipos de instância com suporte. SR-IOV é um método de virtualização de dispositivo que fornece maior desempenho de E / S e menor utilização da CPU quando comparado às interfaces de rede virtualizadas tradicionais. A rede aprimorada fornece largura de banda mais alta, desempenho de pacote por segundo (PPS) mais alto e latências entre instâncias consistentemente mais baixas. Não há cobrança adicional pelo uso de rede avançada.

Elastic Fabric Adapter

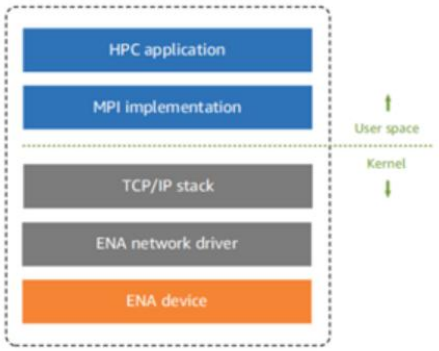
Um Elastic Fabric Adapter (EFA) é um dispositivo de rede que você pode conectar à sua instância do Amazon EC2 para acelerar o High Performance Computing (HPC) e os aplicativos de aprendizado de máquina. O EFA permite que você alcance o desempenho do aplicativo de um cluster HPC local, com a escalabilidade, flexibilidade e elasticidade fornecidas pela Nuvem AWS. O EFA fornece latência mais baixa e mais consistente e maior rendimento do que o transporte TCP tradicionalmente usado em sistemas HPC baseados em nuvem. Ele aprimora o desempenho da comunicação entre instâncias, que é crítica para o dimensionamento de HPC e aplicativos de aprendizado de máquina. Ele é otimizado para funcionar na infraestrutura de rede AWS existente e pode ser dimensionado dependendo dos requisitos do aplicativo.

Questão: Uma empresa planeja implantar um aplicativo em uma frota de instâncias do Amazon EC2 com computação de alto desempenho. Essas instâncias EC2 usam um Elastic Fabric Adapter (EFA) para aprimorar o desempenho e o rendimento. Quais das seguintes afirmações são corretas? (Selecione dois)

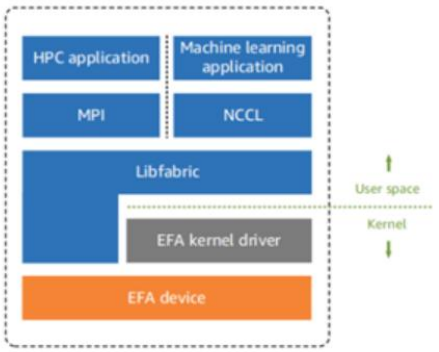
R1: Create EFA in one subnet in a VPC since it is not moved into another subnet after creation

R2: Ensure to enable Amazon CloudWatch metrics to monitor EFAs in real time

O EFA oferece suporte à funcionalidade de bypass do sistema operacional que permite que os aplicativos HPC se comuniquem diretamente com o hardware de interface de rede que oferece baixa latência e alto rendimento. O aplicativo HPC também usa uma interface de passagem de mensagens com o transporte de rede do sistema.



Traditional HPC software stack in EC2



HPC software stack in EC2 with EFA

Differences between EFAs and ENAs

Os Elastic Network Adapters (ENAs) fornecem recursos de rede IP tradicionais necessários para dar suporte à rede VPC. Os EFAs fornecem todos os recursos de rede IP tradicionais dos ENAs e também oferecem suporte a recursos de bypass do sistema operacional. O bypass do SO permite que os aplicativos HPC e de aprendizado de máquina ignorem o kernel do sistema operacional e se comuniquem diretamente com o dispositivo EFA.

Discount Plan

- *Compute Saving plans* also offer flexibility, but the maximum cost reduction is 66%.
- *EC2 saving plan*, you can save up to 72%. This plan applies to all of your EC2 instances using the same instance family. It also offers the flexibility of changing sizes of EC2 instances.
- *Reserved instances*, you can change the instance size. But the maximum save is up to 54%.
- *Dedicated Instances* are Amazon EC2 instances that run in a virtual private cloud (VPC) on hardware that's dedicated to a single customer. They cannot reduce the costs.

Spot Instance

Você lançou 9 instâncias pontuais para uma carga de trabalho específica em sua conta AWS. O preço do seu lance foi de $ 0,07 por hora, e o preço à vista no momento do lançamento foi de $ 0,06 por hora. Após 1,5 hora, o preço à vista sobe para US $ 0,08 a hora. Qual é o custo incorrido? R: $0.54

Instâncias pontuais são aquelas para as quais um usuário deve fazer uma oferta no portal da AWS. Se o preço do lance for maior do que o preço Amazon (ou seja, preço à vista), as instâncias à vista serão automaticamente concedidas. O usuário seria cobrado com base no preço 'spot' em vez do preço 'lance'.

Se o preço do lance for inferior ao preço Amazon (ou seja, preço à vista), as instâncias à vista serão canceladas. Agora, se um usuário tiver uma instância spot em execução e de repente o preço spot subir, a Amazon cancela automaticamente a instância e o usuário não é cobrado pelos minutos extras (arredondados para uma hora). Isso é chamado de encerramento do Amazon da instância local.

Em um segundo caso, se um usuário tiver uma instância spot em execução (quando o preço do lance for maior que o preço spot e o usuário receber a 'instância spot' no preço 'spot') e o próprio usuário encerrar voluntariamente o instância spot, o usuário é cobrado até o minuto em que usa a instância spot. Isso é chamado de rescisão voluntária do usuário da instância local. Com a introdução acima, podemos prosseguir com o cálculo abaixo.

In the first hour,

Bid price = $0.07

Spot price = $0.06

Portanto, o usuário recebe a instância spot. Agora, o preço das instâncias '9' para a primeira hora seria = $ 0,06 * 9 = $ 0,54

In the second hour (i.e., for 0.5 hour)

Bid price = $0.07

Spot price = $0.08

Agora o preço spot é maior do que o preço de oferta que terminará na instância spot sendo encerrada pela Amazon, e o usuário não paga qualquer valor pelas instâncias durante 0,5 hora em que as instâncias foram executadas.

Therefore the total payable amount by the user for '9' instances is = $0.54.

EC2 RI Utilization % vs EC2 RI Coverage %

Uma empresa Fintech de médio porte está usando a Organização AWS para gerenciar várias contas AWS criadas para cada departamento. Cada uma das contas adquiriu uma instância reservada e está executando aplicativos da web em uma combinação de pool sob demanda e instância reservada. Uma política IAM padrão é configurada para todas as contas. Devido aos altos custos recorrentes, a Gerência nomeou você como um consultor da AWS para sugerir recomendações para reduzir custos. Após a análise, você sugeriu comprar mais instância reservada em comparação com o uso da instância EC2 On-Demand. Como você justificaria suas recomendações à administração?

R: Use Organization member account owners to create RI coverage budgets for their individual accounts in an organization & receive SNS alert once the threshold is below 50%.

The Reserved Instance Utilization and Coverage reports are not the same.

EC2 RI Utilization% oferece dados relevantes para identificar e agir em oportunidades para aumentar a eficiência de uso de sua instância reservada. É calculado dividindo as horas de uso da instância reservada pelo total de horas adquiridas da instância reservada.

EC2 RI Coverage% mostra quanto do uso geral da instância é coberto por instâncias reservadas. Isso permite que você tome decisões informadas sobre quando comprar ou modificar uma instância reservada para garantir a cobertura máxima. É calculado dividindo as horas de uso da instância reservada pelo total de horas EC2 On-Demand e da instância reservada.

O orçamento de cobertura do RI informa o número de instâncias que fazem parte da instância reservada. Isso ajuda você a receber um alerta quando o número de instâncias cobertas pela reserva cair abaixo de 50% do número de instâncias iniciadas. Este relatório pode identificar a instância que está em execução consistente usando a instância On-Demand e pode ser convertido em instância reservada para economia de custos. Os proprietários de contas de membros da Organização AWS podem criar um orçamento para contas individuais. A conta mestre da organização AWS paga pelo uso incorrido por todas as contas na organização.

Links Úteis:

https://docs.aws.amazon.com/pt_br/AWSEC2/latest/UserGuide/AMIs.html

https://aws.amazon.com/pt/premiumsupport/knowledge-center/ec2-server-refused-our-key/

https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/troubleshooting-windows-instances.html#rdp-issues

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-metadata.html

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/efa.html

➢ Pontos de Atenção

1. Você está construindo um protótipo para um site de notícias sobre criptomoedas de uma pequena startup. O site será implantado em uma instância do Spot EC2 Linux e usará o Amazon Aurora como seu banco de dados. Você solicitou uma instância spot a um preço máximo de $ 0,04/hora que foi cumprido imediatamente e, após 90 minutos, o preço spot aumenta para $ 0,06/hora e então sua instância foi encerrada pela AWS. Nesse cenário, qual seria o custo total de execução de sua instância spot? R: $0.06

2. Uma empresa está hospedando instâncias EC2 que estão em um ambiente de não produção e processando cargas em lote não prioritárias, que podem ser interrompidas a qualquer momento. Qual é a melhor opção de compra de instância que pode ser aplicada às suas instâncias EC2 neste caso? R: Spot Instances

3. Você é arquiteto de soluções de uma grande rede de TV. Eles têm um aplicativo da web em execução em oito instâncias do Amazon EC2, consumindo cerca de 55% dos recursos em cada instância. Você está usando o Auto Scaling para garantir que oito instâncias estejam em execução o tempo todo. O número de solicitações que este aplicativo processa são consistentes e não apresentam picos. Seu gerente o instruiu a garantir alta disponibilidade desse aplicativo da web em todos os momentos para evitar qualquer perda de receita. Você deseja que a carga seja distribuída uniformemente entre todas as instâncias. Você também deseja usar a mesma Amazon Machine Image (AMI) para todas as instâncias EC2. Como você conseguirá fazer isso? R: Deploy four EC2 instances in one Availability Zone and four in another availability zone in the same region behind an Amazon Elastic Load Balancer.

4. Você está trabalhando para uma grande empresa financeira e foi instruído a configurar um host bastião Linux. Isso permitirá o acesso às instâncias do Amazon EC2 em execução em seu VPC. Para fins de segurança, apenas os clientes que se conectam a partir do endereço IP público externo corporativo 175.45.116.100 devem ter acesso SSH ao host. Qual a melhor opção que atende a necessidade do cliente? R: Security Group Inbound Rule: Protocol – TCP. Port Range – 22, Source 175.45.116.100/32

5. Você está automatizando a criação de instâncias EC2 em seu VPC. Portanto, você escreveu um script python para acionar a API do Amazon EC2 para solicitar 50 instâncias do EC2 em uma única zona de disponibilidade. No entanto, você percebeu que, após 20 solicitações bem-sucedidas, as solicitações subsequentes falharam. Qual poderia ser a razão para esse problema e como você o resolveria? R: Há um limite flexível (soft limit) de 20 instâncias por região, por isso as solicitações subsequentes falharam. Basta enviar o formulário de aumento de limite para a AWS e tentar novamente as solicitações com falha depois de aprovadas.

6. Uma empresa está usando um script de shell personalizado para automatizar a implantação e o gerenciamento de suas instâncias EC2. O script usa vários comandos CLI da AWS, como revoke-security-group-ingress, revoke-security-group-egress, run-schedule-instances e muitos outros. No script de shell, o que o comando revoke-security-group-ingress faz? R: Removes one or more ingress rules from a security group.

7. Seu gerente de TI o instruiu a configurar um host bastião da maneira mais barata e segura e que você deve ser a única pessoa que pode acessá-lo via SSH. Qual das etapas a seguir satisfaria a solicitação do seu gerente de TI? R: Set up a small EC2 instance and a security group which only allows access on port 22 via your IP address

8. Você está trabalhando para uma importante consultoria de TI e um de seus clientes perguntou a você como proteger adequadamente a infraestrutura da AWS. Eles têm um VPC com duas instâncias de EC2 sob demanda com endereços Elastic IP que recentemente sofreram ataques de força bruta SSH pela Internet. Sua equipe de segurança de TI identificou os endereços IP de origem desses ataques. Qual das opções a seguir é a maneira mais rápida de corrigir essa vulnerabilidade de segurança? R: Block the IP addresses in the Network Access Control List

9. Um aplicativo que realiza análises estatísticas de dados meteorológicos recebe arquivos uma vez por semana. Ele assimila os dados desses arquivos com os dados previamente coletados por meio de seus algoritmos e publica um relatório no final de cada mês. Em horários não especificados durante a semana, os resultados provisórios precisam ser disponibilizados aos meteorologistas em minutos. Qual arquitetura atenderá aos requisitos de disponibilidade de dados para a solução com o menor custo e o código de aplicativo mais simples? R: Process the data on EC2 and hibernate the instance until new data files arrive or an interim results request is made

10. You are a Solutions Architect in your company where you are tasked to set up a cloud infrastructure. In the planning, it was discussed that you will need two EC2 instances which should continuously run for three years. The CPU utilization of the EC2 instances is also expected to be stable and predictable. Which is the most cost-efficient Amazon EC2 Pricing type that is most appropriate for this scenario? R: Reserved Instances

11. Você está trabalhando como arquiteto de soluções para um fabricante aeroespacial que usa intensamente a AWS. Eles estão executando um cluster de aplicativos multicamadas que abrange vários servidores para seu modelo de simulação de vento e como isso afeta seu design de asa de última geração. Atualmente, você está enfrentando uma lentidão em seus aplicativos e, após uma investigação mais aprofundada, descobriu-se que isso se deve a problemas de latência. Qual dos seguintes recursos do EC2 você deve usar para otimizar o desempenho de um cluster de computação que requer baixa latência de rede? R: Placement Groups

12. The media company that you are working for has a video transcoding application running on Amazon EC2. Each EC2 instance polls a queue to find out which video should be transcoded, and then runs a transcoding process. If this process is interrupted, the video will be transcoded by another instance based on the queuing system. This application has a large backlog of videos which need to be transcoded. Your manager would like to reduce this backlog by adding more EC2 instances, however, these instances are only needed until the backlog is reduced. In this scenario, which type of Amazon EC2 instance is the most cost-effective type to use without sacrificing performance? R: Spot instances

13. You are unable to connect to your new EC2 instance via SSH from your home computer, which you have recently deployed. However, you were able to successfully access other existing instances in your VPC without any issues. Which of the following should you check and possibly correct to restore connectivity? R: Configure the Security Group of the EC2 instance to permit ingress traffic over port 22 from your IP.

14. You are a Solutions Architect of a tech company. You are having an issue whenever you try to connect to your newly created EC2 instance using a Remote Desktop connection from your computer. Upon checking, you have verified that the instance has a public IP and the Internet gateway and route tables are in place. What else should you do for you to resolve this issue? R: You should adjust the security group to allow traffic from port 3389

15. You are building a microservices architecture in which a software is composed of small independent services that communicate over well-defined APIs. In building large-scale systems, fine-grained decoupling of microservices is a recommended practice to implement. The decoupled services should scale horizontally from each other to improve scalability. What is the difference between Horizontal scaling and Vertical scaling? R: Vertical scaling means running the same software on bigger machines which is limited by the capacity of the individual server. Horizontal scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.

16. The company that you are working for has instructed you to create a cost-effective cloud solution for their online movie ticketing service. Your team has designed a solution of using a fleet of Spot EC2 instances to host the new ticketing web application. You requested a spot instance at a maximum price of $0.06/hr which has been fulfilled immediately. After 45 minutes, the spot price increased to $0.08/hr and then your instance was terminated by AWS. What was the total EC2 compute cost of running your spot instances? R: $0.00

17. A well-known liquor company has a legacy application which needs to be transferred to the AWS cloud. The legacy application has a dependency on the license which is based on its media access control (MAC) address. They will launch the application in an on-demand EC2 instance. The company has hired you to assist them in this

transition. In this scenario, what can you do to ensure that the MAC address of the EC2 instance will not change even if the instance is restarted or rebooted? R: Provision an ENI with a fixed MAC address.

18. Your IT Manager asks you to create a decoupled application whose process includes dependencies on EC2 instances and servers located in your company's on-premises data center. Which of these options are you least likely to recommend as part of that process? R: SQS polling from an EC2 instance using IAM user credentials

19. You are working for a weather station in Asia with a weather monitoring system that needs to be migrated to AWS. Since the monitoring system requires a low network latency, high network throughput, you decided to launch your EC2 instances to a cluster placement group. However, when you try to add new instances to the new placement group, you receive an 'insufficient capacity error'. How will you fix this issue? R: Stop and restart the instances in the Placement Group and then try the launch again.

20. A large Philippine-based Business Process Outsourcing company is building a two-tier web application in their VPC to serve dynamic transaction-based content. The data tier is leveraging an Online Transactional Processing (OLTP) database but for the web tier, they are still deciding what service they will use. What AWS services should you leverage to build an elastic and scalable web tier? R: Elastic Load Balancing, Amazon EC2, and Auto Scaling

21. The IT Operations team of your company wants to retrieve all of the Public IP addresses assigned to a running EC2 instance via the Instance metadata. Which of the following URLs will you use? R: http://169.254.169.254/latest/meta-data/public-ipv4

22. The game development company that you are working for has an Amazon VPC with a public subnet. It has 4 EC2 instances that are deployed in the public subnet. These 4 instances can successfully communicate with other hosts on the Internet. You launch a fifth instance in the same public subnet, using the same AMI and security group configuration that you used for the others. However, this new instance cannot be accessed from the internet unlike the other instance. What should you do to enable access to the fifth instance over the Internet? R: Assign an Elastic IP address to the fifth instance.

23. You have a web application hosted in an On-Demand EC2 instance in your VPC. You are creating a shell script that needs the instance's public and private IP addresses. What is the best way to get the instance's associated IP addresses which your shell script can use? R: By using a Curl or Get Command to get the latest metadata information from http://169.254.169.254/latest/meta-data/

24. A company is using hundreds of AWS resources in multiple AWS regions. They require a way to uniquely identify all of their AWS resources that will allow them to specify a resource unambiguously across all of AWS, such as in IAM policies, Amazon Relational Database Service (Amazon RDS) tags, and API calls. Which of the following is the most suitable option to use in this scenario? R: Amazon Resource Name

25. There is a new compliance rule in your company that audits every Windows and Linux EC2 instances each month to view any performance issues. They have more than a hundred EC2 instances running in production, and each must have a logging function that collects various system details regarding that instance. The SysOps team will periodically review these logs and analyze their contents using AWS Analytics tools, and the result will need to be retained in an S3 bucket. In this scenario, what is the most efficient way to collect and analyze logs from the instances with minimal effort? R: Install the unified CloudWatch Logs agent in each instance which will automatically collect and push data to CloudWatch Logs. Analyze the log data with CloudWatch Logs Insights.

26. You are a Solutions Architect working for a large multinational investment bank. They have a web application that requires a minimum of 4 EC2 instances to run to ensure that it can cater to its users across the globe. You are instructed to ensure fault tolerance of this system. Which of the following is the best option? R: Deploy an Auto Scaling group with 2 instances in each of 3 Availability Zones behind an Application Load Balancer.

27. Você foi encarregado de replicar seu VPC de produção em outra região para fins de recuperação de desastres. Parte do seu ambiente depende de instâncias EC2 com software pré-configurado. Quais etapas você executaria para configurar as instâncias em outra região? R: Create AMIs of the instances and copy them to the new Region for deployment.

28. Você está gerenciando um conjunto de aplicativos em sua rede local que usa endereços IP confiáveis que seus parceiros e clientes colocaram na lista de permissões em seus firewalls. Há um requisito para migrar esses aplicativos para a AWS sem exigir que seus parceiros e clientes alterem suas listas de permissões de endereços IP. R: Create a Route Origin Authorization (ROA) then once done, provision and advertise your whitelisted IP address range to your AWS account.

29. You are running an EC2 instance store-based instance. You shut it down and then start the instance. You noticed that the data which you have saved earlier is no longer available. What might be the cause of this? R: The EC2 instance was using instance store volumes, which are ephemeral and only live for the life of the instance.

30. A financial application that calculates accruals, interests, and other data is hosted on a fleet of Spot EC2 instances that are configured with Auto Scaling. The application is used by an external reporting application that provides the total calculation for each user account and transaction. You used CloudWatch to automatically monitor the EC2 instance without manually checking the server for high CPU Utilization or crashes. What is the time period of data that Amazon CloudWatch receives and aggregates from EC2 by default? R: **Five minutes**

31. A startup company wants to launch a fleet of EC2 instances on AWS. Your manager wants to ensure that the Java programming language is installed automatically when the instance is launched. In which of the below configurations can you achieve this requirement? R: **User data**

32. Você precisa implantar um aplicativo de computação de alto desempenho (HPC) e aprendizado de máquina em instâncias AWS Linux EC2. O desempenho da comunicação entre instâncias é muito crítico para o aplicativo. Você deseja conectar um dispositivo de rede à instância para que o desempenho da computação possa ser bastante aprimorado. Qual das opções a seguir pode obter o melhor desempenho? R: Configure Elastic Fabric Adapter (EFA) in the instance.

33. A equipe de desenvolvimento está trabalhando em um novo aplicativo para o qual lançará uma instância EC2. Para diminuir o tempo de inicialização da instância EC2, eles querem que você pré-aqueça a instância e a mantenha pronta para inicialização com todos os patches e software necessários. Qual das seguintes opções pode ser feita para atender a esse requisito? R: Launch the Amazon EC2 instance with an Amazon EBS root volume and enable Hibernate. Para pré-aquecer a instância do EC2, o EC2 Hibernate pode ser usado. A instância precisa ser iniciada com um volume raiz do Amazon EBS. **Além disso, você não pode hibernar uma instância em um grupo Auto Scaling ou usado pelo Amazon ECS.**

34. Você faz parte da equipe de TI de uma seguradora. Você tem 4 instâncias M5.large EC2 usadas para computar alguns dados de seus serviços principais. A quantidade de uso dessas instâncias tem sido muito consistente. Portanto, você prevê que não aumentará nos próximos dois ou três anos. No entanto, seu CFO está perguntando se há uma maneira de reduzir custos nas instâncias do EC2. O único requisito será poder alterar o tamanho das instâncias. O que você sugere para obter a redução máxima de custo? R: Purchase an EC2 instance saving plan.

35. Um arquiteto de soluções está projetando a arquitetura de nuvem de um novo aplicativo que está sendo implantado na nuvem AWS. Este aplicativo possui um conjunto de instâncias que devem ser colocadas em um grupo de veiculações. As instâncias devem ser colocadas em racks distintos, com cada rack tendo sua própria rede e fonte de alimentação. Qual dos designs a seguir fornece a maior disponibilidade para o aplicativo? R: Place the instances into a spread placement group that can span multiple Availability Zones in the same Region.

A spread placement group supports a maximum of seven running instances per Availability Zone. For example, in a Region with three Availability Zones, a total of 21 instances (7 x 3) in the group (seven per zone). If any plans to start the eighth instance in the same AZ and the same spread placement group, the instance will not launch.

36. Sua equipe de desenvolvimento deseja usar instâncias EC2 para hospedar seus aplicativos e servidores da web. No espaço de automação, eles desejam que as Instâncias sempre baixem a versão mais recente dos servidores Web e de aplicativos quando ativados. Como arquiteto, o que você recomendaria para este cenário? R: Ask the Development team to create scripts which can be added to the User Data section when the instance is launched.

37. You work as an architect for a company. An application will be deployed on a set of EC2 instances in a private subnet of VPC. You need to ensure that IT administrators can securely administer the instances in the private subnet. How can you accomplish this? R: Create a bastion host in the public subnet. Make IT admin staff use this as a jump server to the backend instances.

38. Você está trabalhando para uma empresa iniciante, trabalhando em um projeto POC, no qual várias instâncias do EC2 são iniciadas para um projeto interno para verificar o desempenho do aplicativo da web. Você precisa pré-aquecer as instâncias do EC2, iniciando-as no modo desejado e passando para o estado de hibernação. Você está procurando por alterações de endereço IP quando as instâncias do EC2 passam do estado de execução para hibernação e de volta para o estado de execução. Qual das seguintes afirmações está correta? R: Only Public IPv4 is allocated with new IP while Private IPv4 and any IPv6 are retained.

AWS Elastic Beanstalk

Serviço para deploy de aplicações automatizado.

O AWS Elastic Beanstalk é um serviço de fácil utilização para implantação e escalabilidade de aplicações e serviços da web desenvolvidos com Java, .NET, PHP, Node.js, Python, Ruby, Go e Docker em servidores familiares como Apache, Nginx, Passenger e IIS.

Basta fazer o upload de seu código e o Elastic Beanstalk se encarrega automaticamente da implementação, desde o provisionamento de capacidade, o balanceamento de carga e a escalabilidade automática até o monitoramento da saúde do aplicativo.

Ao mesmo tempo, você mantém total controle sobre os recursos da AWS que possibilitam a operação do seu aplicativo e pode acessar os recursos subjacentes a qualquer momento.

Não há custos adicionais pelo Elastic Beanstalk – você só paga pelos recursos da AWS necessários para executar e armazenar seus aplicativos.

Características

- Deploy na AWS
- Interativo/Fácil
- Abstração -> EC2, RDS, ELB, AS

Tipos de Deploy

All at once: Implante a nova versão em todas as instâncias simultaneamente. Todas as instâncias em seu ambiente estão fora de serviço por um curto período de tempo enquanto a implantação ocorre.

Rolling: Implante a nova versão em lotes

Rolling with additional batch: começa implantando o código do seu aplicativo em um único lote de instâncias EC2 recém-criadas. Depois que a implantação é bem-sucedida no primeiro lote de instâncias, o código do aplicativo é implantado nas instâncias restantes em lotes até que o último lote de instâncias permaneça.

Immutable: Infraestrutura imutável refere-se a servidores (ou VMs) que nunca são modificados após a implantação. Com um paradigma de infraestrutura imutável, os servidores funcionam de forma diferente. Não queremos mais atualizar os servidores locais. Em vez disso, queremos garantir que um servidor implantado permaneça intacto, sem alterações.

Blue-green: A implantação do verde azulado é um modelo de lançamento de aplicativo que transfere gradualmente o tráfego do usuário de uma versão anterior de um aplicativo ou microsserviço para um novo lançamento quase idêntico - ambos em execução na produção.

Traffic-splitting: A implantação permite que você monitore a integridade de sua nova versão do aplicativo em uma porcentagem configurável do tráfego de produção antes de concluir a implantação. No caso de falhas de implantação, uma implantação de divisão de tráfego aciona um mecanismo de reversão automática.

➢ **Pontos de Atenção**

1. Sua empresa está com pressa de implantar seu novo aplicativo da web escrito em NodeJS para AWS. Como arquiteto de soluções da empresa, você foi designado para fazer a implantação sem se preocupar com a infraestrutura subjacente que executa o aplicativo. Qual serviço você usará para implantar e gerenciar facilmente seu novo aplicativo da web na AWS? R: AWS Elastic Beanstalk

2. An online shopping platform has been deployed to AWS using Elastic Beanstalk. They simply uploaded their Node.js application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring. Since the entire deployment process is automated, the DevOps team is not sure where to get the application log files of their shopping platform. In Elastic Beanstalk,

where does it store the application files and server log files? R: Application files are stored in S3. The server log files can also optionally be stored in S3 or in CloudWatch Logs.

3. Uma empresa Startup está lançando um aplicativo de três camadas com a plataforma Multicontainer Docker. Este aplicativo precisa ser integrado à instância do banco de dados Amazon RDS. O aplicativo será iniciado usando o AWS Elastic Beanstalk. Como consultor da AWS para esta empresa, você precisa projetar o ambiente para implantação azul / verde e arquitetura desacoplada no ambiente de produção. O que você recomendaria para integrar o banco de dados Amazon RDS ao AWS Elastic Beanstalk? R: Inicie a instância do Amazon RDS fora do ambiente AWS Elastic Beanstalk, armazenando a string de conexão no bucket S3.

Amazon Elastic Container Service (Amazon ECS)

Serviço de Orquestração de Containers totalmente gerenciado, faz algo parecido com o K8S

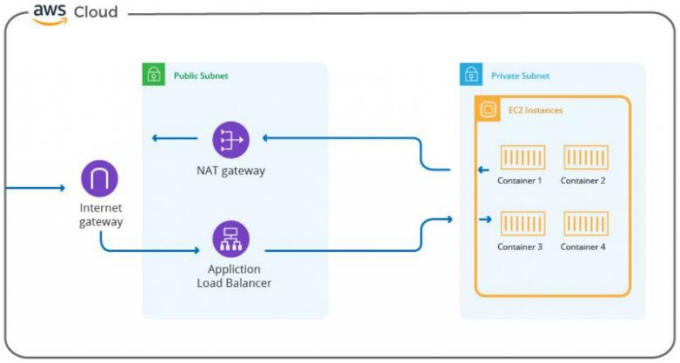
O Amazon ECS facilita a implantação, o gerenciamento e a escala de contêineres Docker que executam aplicativos, serviços e processos em lote.

O Amazon ECS coloca contêineres em seu cluster com base em suas necessidades de recursos e é integrado a recursos familiares como Elastic Load Balancing, grupos de segurança EC2, volumes EBS e funções IAM

O Elastic Container Service (ECS) é um serviço de gerenciamento de contêineres altamente escalonável e rápido. Gerenciar contêineres e executar operações como start/stop é muito fácil no ECS

Os contêineres no ECS são definidos em uma definição de task dentro de um serviço e o serviço é uma configuração que executa e mantém um número especificado de tasks em um cluster

As tarefas podem ser executadas em uma infraestrutura sem servidor gerenciada pelo AWS Fargate ou em um cluster de instâncias do Amazon EC2 gerenciado pelo usuário



O diagrama da arquitetura acima mostra que existem duas sub-redes, pública e privada. As instâncias EC2 são criadas como parte de um cluster ECS que pertence a uma sub-rede privada e um balanceador de carga e um gateway NAT pertencem a uma sub-rede pública

As solicitações de entrada são roteadas por meio de um balanceador de carga para a instância EC2 e, em seguida, são redirecionadas para o contêiner/task, enquanto a conexão de saída da task/contêiner roteia por meio da instância EC2 e, em seguida, do gateway NAT vai para a Internet

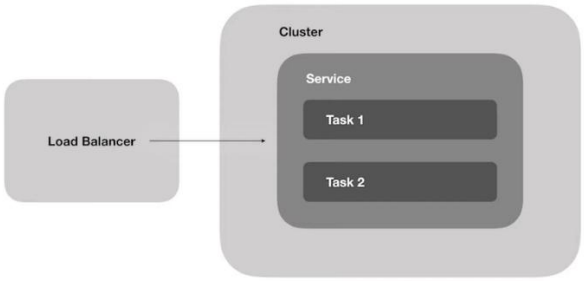
Conceitos

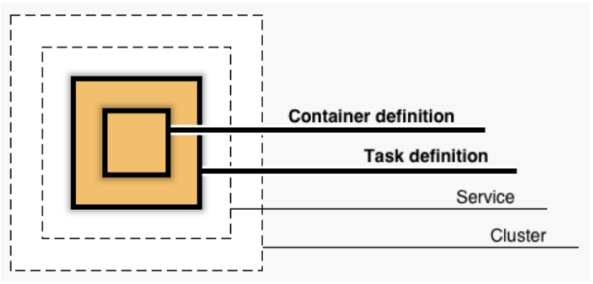
Cluster: Conjunto de máquinas logicamente agrupadas aonde os containers irão rodar. Há 2 tipos de gerenciamento. para criar um cluster é possível usar EC2 ou Fargate

Task Definition: Uma Task Definition é uma especificação (um template) de como uma tarefa deverá ser executada, ou seja, quais containers serão utilizados, recursos computacionais, volumes, entre outros

Task: É quando, a partir de uma Task Definition, uma tarefa em si é instanciada, ou seja, ela começa ser executada de fato, levantando assim containers

Service: O service é um agrupador de tarefas, ou seja, a partir dele podemos definir quantas tarefas deverão ser executadas simultaneamente. Se alguma tarefa, por alguma razão falhar, o scheduler do serviço criará uma nova. O service também é responsável por configurar serviços de rede, como se conectar com um Load Balancer



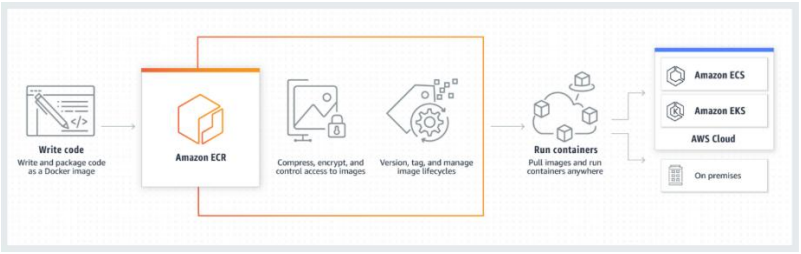


Integrações: Integra com EC2, IAM, Secret Menager, Cloud Watch, ELB. Integra com ferramentas de Devops

Amazon ECR

O Amazon Elastic Container Registry (ECR) é um registro (repositório) de contêiner totalmente gerenciado que facilita o armazenamento, o gerenciamento, o compartilhamento e a implantação de imagens e artefatos de contêiner em qualquer lugar

O Amazon ECR elimina a necessidade de operar os seus próprios repositórios de contêiner ou de preocupar-se sobre a escalabilidade da infraestrutura subjacente. O Amazon ECR hospeda suas imagens em uma arquitetura altamente disponível e de alto desempenho, permitindo que você as implante de maneira confiável em seus aplicativos de contêiner



O Amazon Elastic Container Registry integra-se ao Amazon EKS, Amazon ECS, AWS Lambda e Docker CLI, permitindo que você simplifique seus fluxos de trabalho de desenvolvimento e produção.

Você pode enviar facilmente suas imagens de contêiner ao Amazon ECR usando o Docker CLI a partir da sua máquina de desenvolvimento, e os serviços integrados da AWS podem extraí-las diretamente para implantações em produção

A publicação de softwares de contêiner requer um único comando dos fluxos de trabalho de CI/CD usados no processo do desenvolvedor de software

Fargate

Você pode optar por executar seus clusters do ECS usando o AWS Fargate, que é computação sem servidor para containers. O Fargate elimina a necessidade de provisionar e gerenciar servidores, permitido que você especifique e pague pelos recursos por aplicativos, além de aumentar a segurança ao conceber aplicativos isolados

O preço é baseado nos recursos de vCPU, memória e armazenamento solicitados 1 para a tarefa ou o pod. As três dimensões são configuráveis de forma independente. 20 GB de armazenamento efêmero estão disponíveis para todas as tarefas e pods do Fargate por padrão, você só paga por qualquer armazenamento adicional que configurar

Criando um Cluster

Há 3 modalidades de Cluster, sendo eles:

- AWS Fargate: é possível criar uma VPC e Subnet sendo opcional pois já vem uma VPC e Subnet por default
- EC2 Linux + Networking: é necessário criar a VPC, Subnets e Auto Scaling group
- EC2 Windows + Networking: é necessário criar a VPC, Subnets e Auto Scaling group

É possível habilitar o CloudWatch Container Insights que é uma solução de monitoramento e solução de problemas para microsserviços e aplicativos em contêineres.

Ele coleta, agrega e resume a utilização de computação, como CPU, memória, disco e rede; e informações de diagnóstico, como falhas de reinicialização do contêiner para ajudá-lo a isolar problemas com seus clusters e resolvê-los rapidamente

Ao lançar um Cluster, suas instâncias de contêiner estão sendo iniciadas e pode levar alguns minutos até que estejam no estado de execução e prontas para acesso

As horas de uso em suas novas instâncias de contêiner começam imediatamente e continuam a acumular até que você as interrompa ou encerre

Criando Task Definition

As definições de tarefa especificam as informações do contêiner para seu aplicativo, como quantos contêineres fazem parte de sua tarefa, que recursos eles usarão, como eles estão vinculados e quais portas de host usarão

A task Definition pode ser de 2 tipos, Fargate ou EC2

- Fargate: o preço é baseado no tamanho da task, requer network AWS VPC. A AWS gerencia a infraestrutura
- EC2: preço baseado em recursos utilizados, múltiplas redes disponíveis

Task Size: O tamanho da tarefa permite que você especifique um tamanho fixo para sua tarefa. O tamanho da tarefa é necessário para tarefas que usam o tipo de inicialização Fargate e é opcional para o tipo de inicialização EC2 ou Externo. As configurações de memória no nível do container são opcionais quando o tamanho da tarefa é definido. O tamanho da tarefa não é compatível com os contêineres do Windows

```
{
    "family": "webserver",
    "containerDefinitions": [
        {
            "name": "web",
            "image": "nginx",
            "memory": "100",
            "cpu": "99"
        },
    ],
    "requiresCompatibilities": [
        "FARGATE"
    ],
    "networkMode": "awsvpc",
    "memory": "512",
    "cpu": "256",
}
```

Amazon ECS Task Definitions

A task definition is required to run Docker containers in Amazon ECS. Some of the parameters you can specify in a task definition include:

- The Docker images to use with the containers in your task
- How much CPU and memory to use with each container
- The launch type to use, which determines the infrastructure on which your tasks are hosted
- Whether containers are linked together in a task
- The Docker networking mode to use for the containers in your task
- (Optional) The ports from the container to map to the host container instance
- Whether the task should continue to run if the container finishes or fails
- The command the container should run when it is started
- (Optional) The environment variables that should be passed to the container when it starts
- Any data volumes that should be used with the containers in the task
- (Optional) The IAM role that your tasks should use for permissions

Criando um Service

Um serviço permite que você especifique quantas cópias de sua definição de tarefa executar e manter em um cluster. Opcionalmente, você pode usar um balanceador de carga Elastic Load Balancing para distribuir o tráfego de entrada para contêineres em seu serviço. O Amazon ECS mantém esse número de tarefas e coordena o agendamento de tarefas com o balanceador de carga. Você também pode usar opcionalmente o Service Auto Scaling para ajustar o número de tarefas em seu serviço

Parameters Defined in Service Definition

```
{
    "cluster": "",
    "serviceName": "",
    "taskDefinition": "",
    "loadBalancers": [
        {
            "targetGroupArn": "",
            "loadBalancerName": "",
            "containerName": "",
            "containerPort": 0
        }
    ],
    "serviceRegistries": [
        {
            "registryArn": "",
            "port": 0,
            "containerName": "",
            "containerPort": 0
        }
    ],
    "desiredCount": 0,
    "clientToken": "",
    "launchType": "EC2",
    "platformVersion": "",
    "role": "",
    "deploymentConfiguration": {
        "maximumPercent": 0,
        "minimumHealthyPercent": 0
    },
    "placementConstraints": [
        {
            "type": "memberOf",
            "expression": ""
        }
    ],
    "networkConfiguration": {
        "awsvpcConfiguration": {
            "subnets": [
                ""
            ],
            "securityGroups": [
                ""
            ],
            "assignPublicIp": "ENABLED"
        }
    },
    "healthCheckGracePeriodSeconds": 0,
    "schedulingStrategy": "REPLICA"
}
```

Monitoring

Q: Can I apply additional security configuration and isolation frameworks to my container instances?

Yes. As an Amazon EC2 customer, you have root access to the operating system of your container instances, enabling you to take ownership of the operating system's security settings as well as load and configure additional software components for security capabilities such as monitoring, patch management, log management and host intrusion detection.

User data parameter

Ao iniciar uma instância de contêiner Amazon ECS, você tem a opção de passar os dados do usuário para a instância. Os dados podem ser usados para executar tarefas comuns de configuração automatizada e até mesmo executar scripts quando a instância é inicializada. Para Amazon ECS, os casos de uso mais comuns para dados do usuário são para passar informações de configuração para o Docker daemon e o Amazon ECS container agent.

Amazon ECS Container Agent

The Amazon ECS-optimized AMI looks for agent configuration data in the `/etc/ecs/ecs.config` file when the container agent starts. You can specify this configuration data at launch with Amazon EC2 user data. For more information about available Amazon ECS container agent configuration variables, see Amazon ECS Container Agent Configuration.

To set only a single agent configuration variable, such as the cluster name, use **echo** to copy the variable to the configuration file:

```
#!/bin/bash
echo "ECS_CLUSTER=MyCluster" >> /etc/ecs/ecs.config
```

If you have multiple variables to write to `/etc/ecs/ecs.config`, use the following heredoc format. This format writes everything between the lines beginning with **cat** and `EOF` to the configuration file.

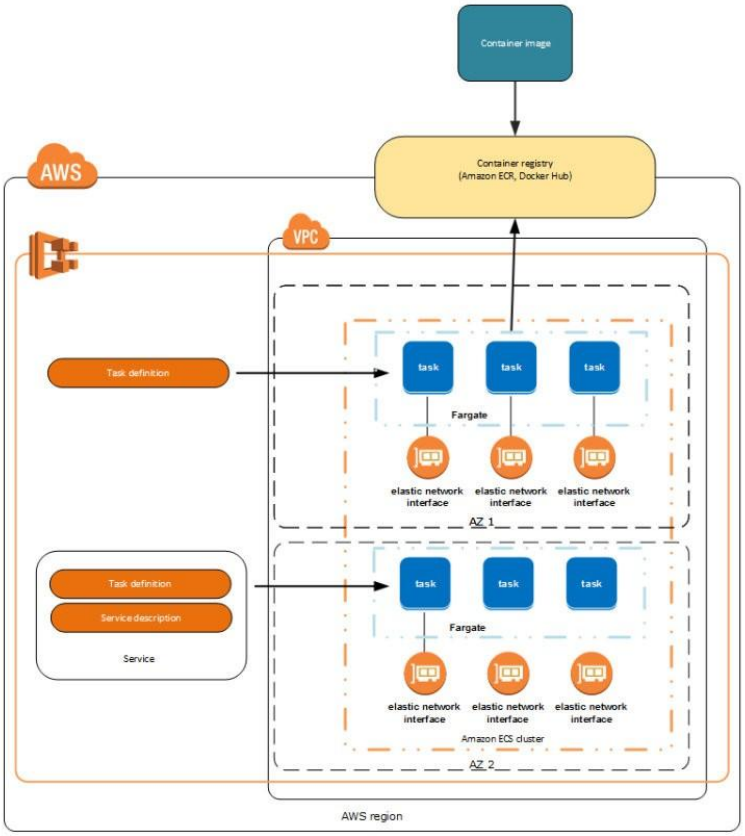
```
#!/bin/bash
cat <<'EOF' >> /etc/ecs/ecs.config
ECS_CLUSTER=MyCluster
ECS_ENGINE_AUTH_TYPE=docker
ECS_ENGINE_AUTH_DATA={"https://index.docker.io/v1/":{"username":"my_name","password":"my_password","email":"email@example.com"}}
ECS_LOGLEVEL=debug
EOF
```

ECS Service Endpoint

- Container instances need access to communicate with the Amazon ECS service endpoint. This can be through an interface VPC endpoint or through your container instances having public IP addresses.

Links Úteis

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-dg.pdf



➢ Pontos de Atenção

1. A new online banking platform has been re-designed to have a microservices architecture in which complex applications are decomposed into smaller, independent services. The new platform is using Docker considering that application containers are optimal for running small, decoupled services. Which service can you use to migrate this new platform to AWS? R: ECS

2. You are launching the AWS ECS instance. You would like to set the ECS container agent configuration during the ECS instance launch. What should you do? R: Set configuration in the user data parameter of ECS instance.

3. Which of the following statement defines task definition? R: JSON template that describes containers which forms your application

4. Your organization is planning to use AWS ECS for docker applications. However, they would like to apply 3rd party monitoring tools on the ECS instances and self-manage these EC2 instances. They approached you asking for a recommendation. What do you suggest? R: Customers will have control over AWS ECS instances and can setup monitoring like a normal EC2 instance.

5. Which of the following is a correct statement concerning ECS instances when accessing the Amazon ECS service endpoint? R: 1) Create an Interface VPC Endpoint for ECS service and attach to VPC subnet's route table in which ECS instances are running. 2) Container instances have public IP addresses.

6. Which of the following are the parameters specified in Service Definition? (choose 3 options) R: 1) Cluster on which to run your service 2) Full ARN of the task definition to run in your service 3) IAM role that allows Amazon ECS to make calls to your load balancer on your behalf

7. You are launching the AWS ECS instance. You would like to set the ECS container agent configuration during the ECS instance launch. What should you do? R: Set configuration in the user data parameter of ECS instance

8. You are working for an organization which is actively using AWS. They have noticed that few AWS ECS clusters are running and they do not know who and when the clusters are created. They tasked you to find out the logs regarding this. What will you do? R: Check CloudTrail logs. Amazon ECS is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon ECS. CloudTrail captures all API calls for Amazon ECS as events, including calls from the Amazon ECS console and from code calls to the Amazon ECS APIs.

9. Your team uses Amazon ECS to manage containers for several micro-services. To save cost, multiple ECS tasks should run at a single container instance. When a task is launched, the host port should be dynamically chosen from the container instance's ephemeral port range. The ECS service should select a load balancer that supports dynamic mapping. Which types of load balancers are appropriate? R:
Application Load Balancer or Network Load Balancer. Because both Application Load Balancer and Network Load Balancer support dynamic mapping. You can configure the ECS service to use the load balancer, and a dynamic port will be selected for each ECS task automatically. With Dynamic mapping, multiple copies of a task can run on the same instance. Classic Load Balancer does not support dynamic mapping. With Classic Load Balancer, you have to define the port mappings on a container instance statically.

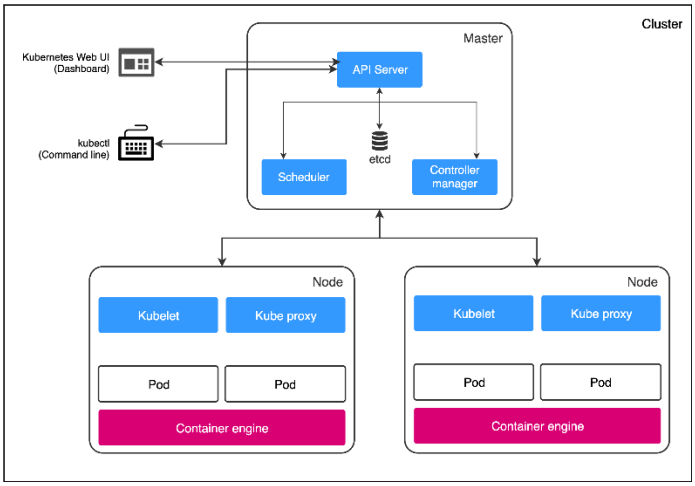
10. Uma empresa planeja implantar um aplicativo de processamento em lote usando contêineres docker. Qual das opções a seguir ajudaria idealmente a hospedar este aplicativo? (SELECIONE DOIS) R: 1. Create a docker image of your batch processing application. 2. Deploy the image as an Amazon ECS task.

Amazon Elastic Kubernetes Service (Amazon EKS)

O que é Kubernetes?

É importante entender o que é Kubernetes antes de prosseguirmos com Amazon ECS vs EKS. **Kubernetes é uma plataforma de orquestração de contêiner de código aberto** que foi originalmente desenvolvida pelo Google e agora é mantida pela Cloud Native Computing Foundation

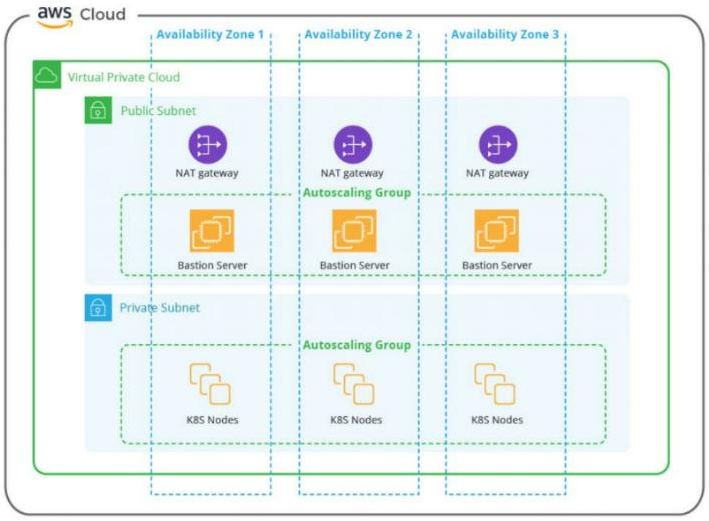
Ele automatiza a implantação, o dimensionamento e o gerenciamento de aplicativos. O Kubernetes oferece a plataforma para agendar e executar contêineres em um grupo de máquinas físicas ou virtuais. Este grupo de máquinas com Kubernetes é conhecido como cluster do Kubernetes



O que é Amazon EKS?

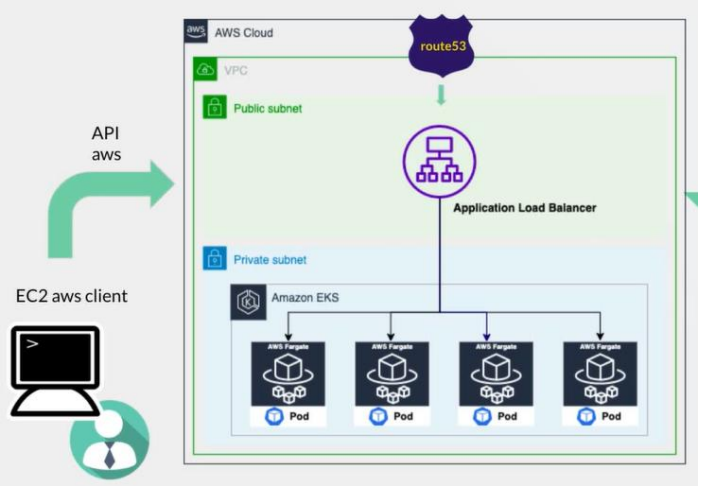
O serviço Elastic Kubernetes, EKS, é um serviço gerenciado que pode ser usado para executar o Kubernetes na AWS. Não há necessidade de instalar, operar e manter o plano de controle ou nós do Kubernetes ao usar o EKS

Para garantir alta disponibilidade, o EKS executa instâncias de plano de controle do Kubernetes em várias zonas de disponibilidade. Quando os nós não estão íntegros, o EKS os substitui automaticamente. EKS fornece escalabilidade e segurança para os aplicativos



No diagrama de arquitetura acima, você pode ver que o cluster EKS se estende por várias zonas de disponibilidade. Ele tem seus nós de trabalho em diferentes zonas de disponibilidade para fornecer alta disponibilidade para o aplicativo

Os pods implantados são colocados nesses nós de trabalho. Os gateways NAT da sub-rede pública permitem o acesso de saída da Internet aos recursos implantados nos nós de trabalho



➢ Pontos de Atenção

Uma empresa iniciante está desenvolvendo um aplicativo baseado em microsserviços usando **orquestração de contêiner de código aberto**. Este aplicativo será integrado com outra nuvem pública. A empresa não tem experiência para provisionar e gerenciar infraestrutura de back-end para configurar este contêiner. Você foi designado para fornecer consultoria para a implantação de contêineres. R: Use Amazon Elastic Kubernetes Service with AWS Fargate launch type. *Amazon Elastic Kubernetes Service can be used to set up open-source Container orchestration like Kubernetes. Amazon Elastic Kubernetes Service can be integrated with other public or private clouds.*

Amazon ECS vs EKS

Orquestradores de Containers

- Controla seus containers
- Features de Auto Scaling
- Gerenciamento de Volumes
- Gerenciamento de Replicas
- Gerenciamento de Rede
- Ajuda em Deploys

Comparação

Networking: O EKS permite até 750 pods por instância, enquanto o ECS acomoda apenas até um máximo de 120 tasks por instância, este é um dos pontos importantes a serem entendidos ao considerar a diferença entre Amazon ECS e EKS

Namespaces: O Kubernetes tem o conceito de namespaces que isola as cargas de trabalho em execução no mesmo cluster, enquanto o ECS não tem esse conceito. Os namespaces oferecem muitas vantagens. Por exemplo, você pode ter um ambiente de Desenvolvimento, Preparação e Produção no mesmo cluster, que pode compartilhar recursos do cluster. Portanto, ao decidir entre Amazon ECS e EKS, é importante considerar isso

Fácil de Usar: O ECS é muito simples e não tem muitos componentes para aprender, enquanto o EKS é mais complexo, pois usa o Kubernetes, que é uma tecnologia vasta para aprender e requer experiência para implantações. O ECS não possui nenhum plano de controle, ao contrário do EKS

Preço e Custos: O Amazon EKS consiste em um plano de controle e infraestrutura, AWS Fargate ou Amazon EC2, para hospedar aplicativos em contêineres. O preço do ECS e do EKS também depende da infraestrutura (AWS Fargate ou Amazon EC2) que está sendo usada para hospedar aplicativos em contêineres. Além disso, é necessário pagar $ 0,10 por hora para cada cluster Amazon EKS para seu plano de controle

Portabilidade e Compatibilidade: Ambos, EKS e ECS, são serviços gerenciados da AWS. ECS é um serviço proprietário da AWS, enquanto o EKS é um serviço Kubernetes como plataforma da AWS. Todos os aplicativos em execução em clusters EKS podem ser executados em qualquer outro cluster Kubernetes com pouca ou nenhuma mudança, enquanto a implantação de aplicativos no ECS significa usar a plataforma de contêiner proprietária da AWS. Pode-se mover facilmente os aplicativos do EKS para o autogerenciado ou cluster Kubernetes de qualquer outro provedor de nuvem. Mas esse não é o caso dos aplicativos implantados no ECS

Parameter	Amazon EKS	Amazon ECS
Open Source.	Yes, Kubernetes is open-source.	No, it is an AWS proprietary service.
Smallest Deployable Entity.	Pod.	Task.
Multi-cloud Integration.	Public and Private cloud integration.	AWS Specific.
Pricing.	Includes an additional cost of the control plane.	One needs to pay only for what is used.
Container Limit.	Up to 750 pods per VM.	Up to 120 per VM.
Community Support.	More.	Less.
Security.	EKS does not support IAM for pods.	ECS supports IAM Roles for Tasks.
Ease of use.	Deployment is complex than ECS.	Easier deployments than EKS.

Elastic Load Balancing

O Elastic Load Balancing distribui automaticamente seu tráfego de entrada em vários destinos, como instâncias EC2, contêineres e endereços IP, em uma ou mais zonas de disponibilidade.

Ele monitora a integridade de seus alvos registrados e roteia o tráfego apenas para os alvos íntegros. O Elastic Load Balancing dimensiona seu balanceador de carga conforme o tráfego de entrada muda ao longo do tempo. Ele pode ser dimensionado automaticamente para a grande maioria das cargas de trabalho.

Benefícios do Balanceador de Carga

Um balanceador de carga distribui cargas de trabalho em vários recursos de computação, como servidores virtuais. Usar um balanceador de carga aumenta a disponibilidade e tolerância a falhas de seus aplicativos.

Você pode adicionar e remover recursos de computação de seu balanceador de carga conforme suas necessidades mudam, sem interromper o fluxo geral de solicitações para seus aplicativos.

Você pode configurar verificações de integridade, que monitoram a integridade dos recursos de computação, para que o balanceador de carga envie solicitações apenas para aqueles que estão íntegros.

Você também pode descarregar o trabalho de criptografia e descriptografia para seu balanceador de carga para que seus recursos de computação possam se concentrar em seu trabalho principal.

Serviços Relacionados

Elastic Load Balancing funciona com os seguintes serviços para melhorar a disponibilidade e escalabilidade de seus aplicativos.

Amazon EC2 - servidores virtuais que executam seus aplicativos na nuvem. Você pode configurar seu balanceador de carga para rotear o tráfego para suas instâncias EC2.

Amazon EC2 Auto Scaling - garante que você esteja executando o número desejado de instâncias, mesmo se uma instância falhar. O Amazon EC2 Auto Scaling também permite que você aumente ou diminua automaticamente o número de instâncias conforme a demanda em suas instâncias muda. Se você ativar o Auto Scaling com Elastic Load Balancing, as instâncias iniciadas pelo Auto Scaling serão registradas automaticamente com o balanceador de carga. Da mesma forma, as instâncias encerradas pelo Auto Scaling são automaticamente canceladas no balanceador de carga.

AWS Certificate Manager - Ao criar um ouvinte HTTPS, você pode especificar certificados fornecidos pelo ACM. O balanceador de carga usa certificados para encerrar conexões e descriptografar solicitações de clientes.

Amazon CloudWatch - permite monitorar seu balanceador de carga e agir conforme necessário.

Amazon ECS - permite que você execute, pare e gerencie contêineres Docker em um cluster de instâncias EC2. Você pode configurar seu balanceador de carga para rotear o tráfego para seus contêineres.

AWS Global Accelerator - melhora a disponibilidade e o desempenho do seu aplicativo. Use um acelerador para distribuir o tráfego entre vários balanceadores de carga em uma ou mais regiões da AWS.

Route53 - fornece uma maneira confiável e econômica de direcionar visitantes para sites, traduzindo nomes de domínio em endereços IP numéricos que os computadores usam para se conectar uns aos outros. Por exemplo, isso traduziria www.example.com no endereço IP numérico 192.0.2.1. A AWS atribui URLs aos seus recursos, como balanceadores de carga. No entanto, você pode querer um URL que seja fácil para os usuários se lembrarem.

AWS WAF - você pode usar o AWS WAF com seu balanceador de carga de aplicativo para permitir ou bloquear solicitações com base nas regras em uma lista de controle de acesso à web (ACL da web).

Como o Elastic Load Balancing Trabalha

Um balanceador de carga aceita tráfego de entrada de clientes e roteia solicitações para seus destinos registrados (como instâncias EC2) em uma ou mais zonas de disponibilidade. O balanceador de carga também monitora a integridade de seus destinos registrados e garante que ele direcione o tráfego apenas para destinos saudáveis. Quando o balanceador de carga detecta um destino não íntegro, ele para de rotear o tráfego para esse destino. Em seguida, ele retoma o roteamento do tráfego para esse destino quando detecta que o destino está íntegro novamente.

Você configura seu balanceador de carga para aceitar o tráfego de entrada especificando um ou mais ouvintes. Um ouvinte é um processo que verifica as solicitações de conexão. Ele é configurado com um protocolo e número de porta para conexões de clientes ao balanceador de carga. Da mesma forma, ele é configurado com um protocolo e número de porta para conexões do balanceador de carga para os destinos.

O Elastic Load Balancing é compatível com os seguintes tipos de balanceadores de carga:

- **Application Load Balancers**: Trabalha na Layer 7 do modelo OZI com os protocolos HTTP/HTTPS conseguindo identificar a origem/destino dos pacotes
- **Network Load Balancers**: Network Layer (TCP) Layer 4
- Gateway Load Balancers
- **Classic Load Balancers**: Também conhecido como Elastic Load Balancer, se tornou um legado, trabalhava na Layer 7 e 4

Configuração de Health Check

É necessário escolher o Ping Protocol, Ping Port e Ping Path (como por exemplo: "/index.html")

Response Timeout: é o tempo de espera de resposta do servidor, excedendo esse limite, o LB identificará o servidor como Unhealthy

Interval: é o intervalo de checagem

Unhealthy Threshold: este valor determina o número de vezes que é necessário o servidor estar Unhealthy para que seja entendido como indisponível

Healthy Threshold: este valor determina o número de checagens necessárias para que o servidor seja entendido como disponível

Zonas de Disponibilidade

Quando você habilita uma Zona de Disponibilidade para seu balanceador de carga, o Elastic Load Balancing cria um nó do balanceador de carga na Zona de Disponibilidade. Se você registrar os destinos em uma Zona de Disponibilidade, mas não habilitar a Zona de Disponibilidade, esses destinos registrados não receberão tráfego. Seu balanceador de carga é mais eficaz quando você garante que cada Zona de disponibilidade habilitada tenha pelo menos um destino registrado.

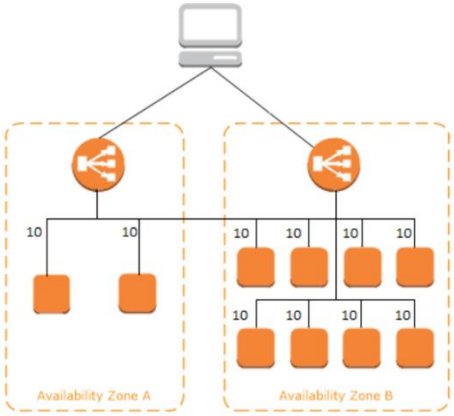
Balanceamento de Carga entre Zonas (Cross-Zone)

Os nós para seu balanceador de carga distribuem solicitações de clientes para destinos registrados. Quando o balanceamento de carga de zona cruzada está habilitado, cada nó do balanceador de carga distribui o tráfego entre os destinos registrados em todas as zonas de disponibilidade habilitadas. Quando o balanceamento de carga de zona cruzada está desabilitado, cada nó do balanceador de carga distribui o tráfego apenas entre os destinos registrados em sua Zona de disponibilidade.

Os diagramas a seguir demonstram o efeito do balanceamento de carga entre zonas. Existem duas zonas de disponibilidade habilitadas, com dois destinos na Zona de disponibilidade A e oito destinos na Zona de disponibilidade B. Os clientes enviam solicitações e o Amazon Route53 responde a cada solicitação com o endereço IP de um dos nós do balanceador de carga. Isso distribui o tráfego de forma que cada nó do balanceador de carga

receba 50% do tráfego dos clientes. Cada nó do balanceador de carga distribui sua parcela do tráfego entre os destinos registrados em seu escopo.

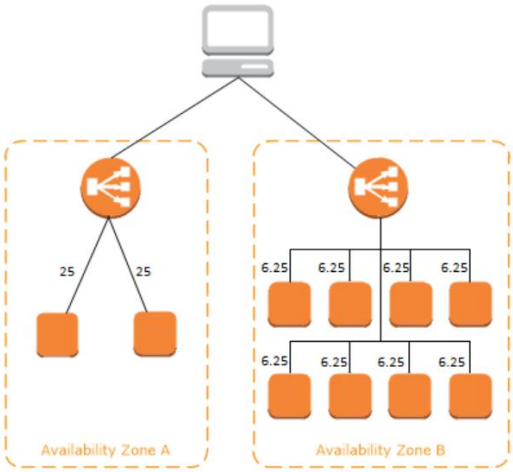
Se o balanceamento de carga de zona cruzada estiver habilitado, cada um dos 10 destinos receberá 10% do tráfego. Isso ocorre porque cada nó do balanceador de carga pode rotear seus 50% do tráfego do cliente para todos os 10 destinos.



Se o balanceamento de carga de zona cruzada estiver desativado:

- Cada um dos dois destinos na Zona de disponibilidade A recebe 25% do tráfego.
- Cada um dos oito destinos na Zona de Disponibilidade B recebe 6,25% do tráfego.

Isso ocorre porque cada nó do balanceador de carga pode rotear seus 50% do tráfego do cliente apenas para destinos em sua Zona de disponibilidade.



Com os balanceadores de carga de aplicativos, o balanceamento de carga de zona cruzada está sempre ativado.

Com os balanceadores de carga de rede e balanceadores de carga de gateway, o balanceamento de carga entre zonas é desabilitado por padrão. Depois de criar o balanceador de carga, você pode habilitar ou desabilitar o balanceamento de carga de zona cruzada a qualquer momento.

Request Routing

Antes de um cliente enviar uma solicitação ao balanceador de carga, ele resolve o nome de domínio do balanceador de carga usando um servidor DNS (Sistema de Nomes de Domínio). A entrada DNS é controlada pela Amazon, porque seus balanceadores de carga estão no domínio amazonaws.com. Os servidores Amazon DNS retornam um ou mais endereços IP para o cliente. Esses são os endereços IP dos nós do balanceador de carga para seu balanceador de carga. Com os Network Load Balancers, o Elastic Load Balancing cria uma interface de rede para cada zona de disponibilidade habilitada. Cada nó do balanceador de carga na Zona de disponibilidade usa essa interface de rede para obter um endereço IP estático. Opcionalmente, você pode associar um endereço Elastic IP a cada interface de rede ao criar o balanceador de carga.

Server Name Indication (SNI)

Server Name Indication (SNI) é uma extensão do protocolo de rede de computador Transport Layer Security (TLS), pelo qual um cliente indica a qual nome de host ele está tentando se conectar no início do processo de handshaking.

[1] Isso permite que um servidor apresente vários certificados no mesmo endereço IP e número de porta TCP e, portanto, permite que vários sites seguros (HTTPS) (ou qualquer outro serviço por TLS) sejam atendidos pelo mesmo endereço IP sem exigir que todos esses sites usem o mesmo certificado. É o equivalente conceitual à hospedagem virtual baseada em nomes HTTP / 1.1, mas para HTTPS. Isso também permite que um proxy encaminhe o tráfego do cliente para o servidor correto durante o handshake TLS / SSL. O nome do host desejado não é criptografado na extensão SNI original, portanto, um intruso pode ver qual site está sendo solicitado.

Temos o prazer de anunciar o suporte a vários certificados TLS em Network Load Balancers usando Server Name Indication (SNI). Agora, você pode hospedar vários aplicativos seguros, cada um com seu próprio certificado TLS, em um único listener de load balancer. Esse recurso permite que aplicativos SaaS e serviços de hospedagem sejam executados atrás do mesmo balanceador de carga, aprimorando a postura de segurança do serviço e simplificando o gerenciamento e as operações.

Antes desse lançamento, os Network Load Balancers ofereciam suporte a apenas um certificado por listener TLS e você precisava usar certificados curinga ou multidomínio (SAN) para hospedar vários aplicativos seguros atrás do mesmo load balancer. Os potenciais riscos de segurança com certificados curinga e a sobrecarga operacional do gerenciamento de certificados multidomínio apresentavam desafios. Com o suporte ao SNI, você pode associar vários certificados a um listener, o que permite que cada aplicativo seguro atrás de um load balancer use seu próprio certificado.

Os Network Load Balancers também oferecem suporte a um algoritmo inteligente de seleção de certificados com SNI. Se o nome de host indicado por um cliente corresponder a vários certificados, o load balancer determinará o melhor certificado a usar considerando vários fatores, incluindo os recursos de TLS do cliente.

O SNI é integrado ao AWS Certificate Manager (ACM) e ao AWS Identity and Access Management (IAM) para o gerenciamento de certificados. Você pode associar até 25 certificados a um load balancer, além de um certificado padrão por listener.

Round-robin

Round-robin é um dos algoritmos empregados por escalonadores de processo e de rede, em computação. Como o termo é geralmente usado, fatias de tempo são atribuídas a cada processo em partes iguais e em ordem circular, manipulando todos os processos sem prioridade.

O agendamento round-robin geralmente emprega tempo compartilhado, dando a cada tarefa um tempo definido chamado quantum. A tarefa é interrompida se esgotado o quantum e retomará de onde parou no próximo agendamento. Sem o tempo compartilhado, tarefas grandes poderiam ser favorecidas em detrimento de tarefas menores.

Configure sessões de perdurabilidade para o Classic Load Balancer

Por padrão, um Classic Load Balancer roteia cada solicitação de forma independente para a instância registrada com a menor carga. No entanto, você pode usar o recurso sticky session (também conhecida como afinidade de sessão), que permite que o load balancer vincule a sessão de um usuário a uma instância específica. Isso garante que todas as solicitações do usuário durante a sessão sejam enviadas para a mesma instância.

O segredo para o gerenciamento de sticky sessions é determinar por quanto tempo o load balancer deve rotear consistentemente a solicitação do usuário para a mesma instância. Se seu aplicativo tiver seu próprio cookie de sessão, você pode configurar o Elastic Load Balancing de forma que o cookie da sessão siga a duração especificada pelo cookie de sessão do aplicativo. Se seu aplicativo não tiver seu próprio cookie de sessão, você poderá configurar o Elastic Load Balancing para criar um cookie de sessão ao especificar sua própria duração de perdurabilidade (aderência).

O Elastic Load Balancing cria um cookie, chamado AWSELB, que é usado para mapear a sessão para a instância.

<u>Requirements</u>

- Um load balancer HTTP/HTTPS.
- Pelo menos uma instância íntegra em cada Zona de disponibilidade.

Elastic Load Balancing Logs

Elastic Load Balancing provides access logs that capture detailed information about requests sent to your load balancer. Each log contains information on when the request was received, the client's IP address, latencies, request paths, and server responses. You can use these access logs to analyze traffic patterns and to troubleshoot issues.

Is possible enable the logs on the ELB with Latency Alarm that sends an email and then investigate the logs whenever there is an issue.

Monitoring

Monitor Your Application Load Balancers

You can use the following features to monitor your load balancers, analyze traffic patterns, and troubleshoot issues with your load balancers and targets.

CloudWatch metrics

You can use Amazon CloudWatch to retrieve statistics about data points for your load balancers and targets as an ordered set of time-series data, known as *metrics*. You can use these metrics to verify that your system is performing as expected. For more information, see CloudWatch Metrics for Your Application Load Balancer.

Access logs

You can use access logs to capture detailed information about the requests made to your load balancer and store them as log files in Amazon S3. You can use these access logs to analyze traffic patterns and to troubleshoot issues with your targets. For more information, see Access Logs for Your Application Load Balancer.

Request tracing

You can use request tracing to track HTTP requests. The load balancer adds a header with a trace identifier to each request it receives. For more information, see Request Tracing for Your Application Load Balancer.

CloudTrail logs

You can use AWS CloudTrail to capture detailed information about the calls made to the Elastic Load Balancing API and store them as log files in Amazon S3. You can use these CloudTrail logs to determine which calls were made, the source IP address where the call came from, who made the call, when the call was made, and so on. For more information, see Logging API Calls for Your Application Load Balancer Using AWS CloudTrail.

Internet-facing

Clients cannot connect to an Internet-facing load balancer

If the load balancer is not responding to requests, check for the following:

Your Internet-facing load balancer is attached to a private subnet

Verify that you specified public subnets for your load balancer. A public subnet has a route to the Internet Gateway for your virtual private cloud (VPC).

A security group or network ACL does not allow traffic

The security group for the load balancer and any network ACLs for the load balancer subnets must allow inbound traffic from the clients and outbound traffic to the clients on the listener ports.

Register a target

Target Type

When you create a target group, you specify its target type, which determines how you specify its targets. After you create a target group, you cannot change its target type.

The following are the possible target types:

`instance`

The targets are specified by instance ID.

`ip`

The targets are specified by IP address.

When the target type is `ip`, you can specify IP addresses from one of the following CIDR blocks:

- The subnets of the VPC for the target group
- 10.0.0.0/8 (RFC 1918)
- 100.64.0.0/10 (RFC 6598)
- 172.16.0.0/12 (RFC 1918)
- 192.168.0.0/16 (RFC 1918)

These supported CIDR blocks enable you to register the following with a target group: ClassicLink instances, instances in a peered VPC, AWS resources that are addressable by IP address and port (for example, databases), and on-premises resources linked to AWS through AWS Direct Connect or a VPN connection.

Important

You can't specify publicly routable IP addresses.

If you specify targets using an instance ID, traffic is routed to instances using the primary private IP address specified in the primary network interface for the instance. If you specify targets using IP addresses, you can route traffic to an instance using any private IP address from one or more network interfaces. This enables multiple applications on an instance to use the same port. Each network interface can have its own security group.

Design Resilient

Você tem um aplicativo da web de duas camadas crítico para os negócios, atualmente implantado em 2 zonas de disponibilidade em uma única região, usando Elastic Load Balancing e Auto Scaling. O aplicativo depende da replicação síncrona na camada do banco de dados. O aplicativo precisa permanecer totalmente disponível, mesmo se um aplicativo AZ ficar off-line repentinamente e o Auto Scaling não puder iniciar novas instâncias no AZ restante. Como o Elastic Load Balancing atual pode ser aprimorado para garantir isso?

R: Deploy in 3 AZ with Auto Scaling, set to handle a minimum 50 percent peak load per zone. Since the requirement states that the application should never go down even if an AZ is not available, we need to maintain 100% availability.

NOTE:

In the question, it is clearly mentioned that "The application needs to remain fully available even if one application AZ goes offline and if Auto Scaling cannot launch new instances in the remaining AZ."

Here you need to maintain 100% availability.

In option B, when you create 3 AZs with a minimum 33% load on each, if any failure occurs in one AZ, then

33% + 33% = 66%

Here you can handle only 66% and the remaining 34% of load is not handled.

But when you select option C, when you create 3 AZs with a minimum 50% load on each, if any failure occurs in one AZ, then

50% + 50% =100%

Here you can handle full load, i.e., 100%.

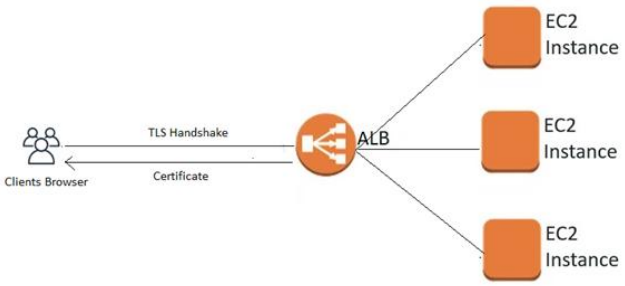
Application Load Balancer vs Classic

Using an Application Load Balancer instead of a Classic Load Balancer has the following benefits:

- Support for path-based routing. You can configure rules for your listener that forward requests based on the URL in the request. This enables you to structure your application as smaller services, and route requests to the correct service based on the content of the URL.
- Support for host-based routing. You can configure rules for your listener that forward requests based on the host field in the HTTP header. This enables you to route requests to multiple domains using a single load balancer.
- Support for routing requests to multiple applications on a single EC2 instance. You can register each instance or IP address with the same target group using multiple ports.
- Support for registering targets by IP address, including targets outside the VPC for the load balancer.
- Support for containerized applications. Amazon Elastic Container Service (Amazon ECS) can select an unused port when scheduling a task and register the task with a target group using this port. This enables you to make efficient use of your clusters.
- Support for monitoring the health of each service independently, as health checks are defined at the target group level and many CloudWatch metrics are reported at the target group level. Attaching a target group to an Auto Scaling group enables you to scale each service dynamically based on demand.
- Access logs contain additional information and are stored in compressed format.
- Improved load balancer performance.

ALB supports Server Name Indication (SNI)

ALB suporta Server Name Indication (SNI), permitindo hospedar vários nomes de domínio com diferentes certificados TLS por trás de um único ALB. Com o SNI, vários certificados podem ser associados a ouvintes no ALB, permitindo que cada aplicativo da web use certificados separados. Os diagramas abaixo mostram o processo que ocorre quando um cliente tenta acessar um site. O navegador do cliente inicia um handshake TLS enviando uma mensagem ClientHello que consiste na versão do protocolo, extensões, conjuntos de criptografia e técnicas de compactação. Com base nos recursos do navegador, o ALB responde com um certificado válido para um nome de domínio do aplicativo da web solicitado.



Protocols & Ciphers

Você está trabalhando como consultor da AWS para um instituto bancário. Eles implantaram uma plataforma de carteira digital para clientes que usam várias instâncias EC2 na região us-east-1. O aplicativo estabelece uma conexão criptografada segura entre clientes e instâncias EC2 para cada transação usando a porta TCP 5810 personalizada.

Devido à crescente popularidade dessa carteira digital, eles observam a carga nos servidores de back-end, resultando em atraso na transação. Para fins de segurança, todos os endereços IP do cliente que acessam este aplicativo devem ser preservados e registrados. A equipe técnica da instituição bancária está em busca de uma solução que dê conta desse atraso e também a solução proposta deve ser compatível com milhões de transações feitas simultaneamente. Qual das opções a seguir é uma opção recomendada para atender a esse requisito?

R: Use Network Load Balancers with SSL certificate. Configure TLS Listeners on this NLB with default security policy consisting of protocols & ciphers.

O Network Load Balancer pode encerrar conexões TLS em vez de instâncias de back-end, reduzindo a carga nesta instância. Com os balanceadores de carga de rede, milhões de sessões simultâneas podem ser estabelecidas sem nenhum impacto na latência, além de preservar o endereço IP do cliente. Para negociar conexões TLS com clientes, o NLB usa uma política de segurança que consiste em protocolos e cifras.

AWS Certificate Manager

Você está trabalhando como consultor da AWS para uma mercearia online. Eles estão usando um aplicativo da web de duas camadas com servidores da web hospedados em VPCs na região us-east-1 e data center local. O balanceador de carga de rede é configurado no front-end para distribuir o tráfego entre esses servidores. Todo o tráfego entre clientes e servidores é criptografado. Eles estão procurando uma solução alternativa para encerrar a conexão TLS neste balanceador de carga de rede para reduzir a carga em servidores back-end.

A equipe de gerenciamento desta loja contratou você para sugerir uma solução para gerenciamento de certificados usado em caso de rescisão de TLS. Qual das opções a seguir é uma opção segura preferencial para provisionar e armazenar certificados a serem usados junto com o Network Load Balancer para encerrar o TLS?

R: Use a single certificate per TLS listener provided by AWS Certificate Manager.

O Network Load Balancer requer um certificado por conexão TLS para criptografar o tráfego entre o cliente e o NLB e encaminhar o tráfego descriptografado para os servidores de destino. Usar o AWS Certificate Manager é uma opção preferencial, pois esses certificados são renovados automaticamente no vencimento.

➢ Pontos de Atenção

1. Você tem um novo aplicativo da web de comércio eletrônico escrito em estrutura Angular que é implementado em uma frota de instâncias EC2 por trás de um Balanceador de Carga de Aplicativo. Você configurou o balanceador de carga para realizar verificações de integridade nessas instâncias do EC2. O que acontecerá se uma dessas instâncias do EC2 falhar nas verificações de integridade? R: O balanceador de carga do aplicativo para de enviar tráfego para a instância que falhou na verificação de integridade

2. Em seu VPC, você tem um balanceador de carga clássico distribuindo tráfego para 2 instâncias EC2 em execução em ap-sudeste-1a AZ e 8 instâncias EC2 em ap-sudeste-1b AZ. No entanto, você notou que metade do tráfego de entrada vai para ap-sudeste-1a AZ, que superutiliza suas 2 instâncias, mas subutiliza as outras 8 instâncias no outro AZ. Qual poderia ser a causa mais provável desse problema? R: Cross-Zone Load Balancing is disabled.

3. Uma empresa de viagens tem um conjunto de aplicativos da web hospedado em um grupo Auto Scaling de instâncias do EC2 sob demanda por trás de um Balanceador de carga de aplicativo que lida com o tráfego de vários domínios da web, como i-love-manila.com, i-love-boracay.com , i-love-cebu.com e muitos outros. Para melhorar a segurança e reduzir o custo geral, você é instruído a proteger o sistema permitindo que vários domínios atendam ao tráfego SSL sem a necessidade de reautenticar e reprovisionar seu certificado sempre que adicionar um novo domínio. Esta migração de HTTP para HTTPS ajudará a melhorar seu SEO e classificação de pesquisa do Google Qual das alternativas a seguir é a solução mais econômica para atender ao requisito acima? R: Carregue todos os certificados SSL dos domínios no ALB usando o console e vincule vários certificados ao mesmo ouvinte seguro em seu balanceador de carga. ALB escolherá automaticamente o certificado TLS ideal para cada cliente usando Server Name Indication (SNI).

4. Você está designado para projetar uma arquitetura altamente disponível na AWS. Você tem dois grupos de destino com três instâncias EC2 cada, que são adicionados a um Balanceador de Carga de Aplicativo. No grupo de segurança da instância EC2, você verificou se a porta 80 para HTTP é permitida. No entanto, as instâncias ainda estão fora de serviço do balanceador de carga. Qual pode ser a causa raiz desse problema? R: The health check configuration is not properly defined.

5. You are working for a large telecommunications company where you need to run analytics against all combined log files from your Application Load Balancer as part of the regulatory requirements. Which AWS services can be used together to collect logs and then easily perform log analysis? R: Amazon S3 for storing ELB log files and Amazon EMR for analyzing the log files.

6. You have a web application hosted on a fleet of EC2 instances located in two Availability Zones that are all placed behind an Application Load Balancer. As a Solutions Architect, you have to add a health check configuration to ensure your application is highly-available. Which health checks will you implement? R: HTTP or HTTPS health check

7. Which AWS Load Balancer types uses a Round-Robin load distribution strategy? R: 1) The NLB does not uses a Round-Robin strategy. 2) The Classic uses a Round-Robin strategy for TCP listeners only. 3) The ALB 1st selects a target based on the routing rule, then uses a Round-Robin strategy to select a node.

8. Um cliente está hospedando o site da empresa em um cluster de servidores da web atrás de um balanceador de carga voltado para o público. O cliente também usa o Amazon Route 53 para gerenciar seu DNS público. Como o cliente deve configurar o registro de apex da zona DNS para apontar para o balanceador de carga? R: Create an A record aliased to the load balancer DNS name.

9. In an AWS Setup of a company, a web-based application has a fleet of 10 EC2 instances. 7 EC2 instances are present in Availability Zone A, whereas 3 EC2 instances in Availability Zone B. The percentage (%) of requests received in Availability Zone B is greater than the percentage (%) of requests in Availability Zone A. What can be done at the architecture level to balance the load across the two availability zones? Please select 3 correct options. R: 1) Use Application Load Balancer to achieve this ability. 2) This can be achieved through cross-zone load balancing. 3) Use Network Load Balancer to achieve his ability.

AWS Fargate

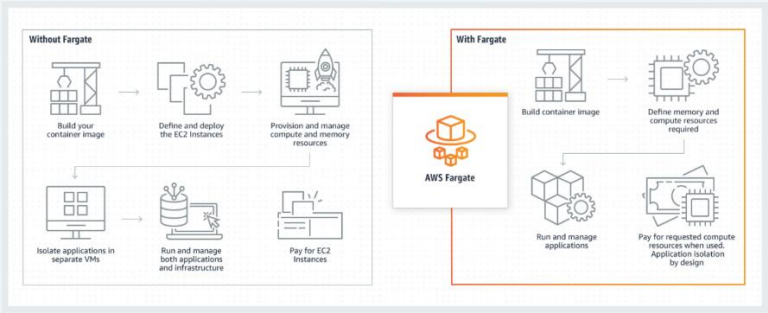
O AWS Fargate é um mecanismo de computação sem servidor para contêineres que funciona com o Amazon Elastic Container Service (ECS) e com o Amazon Elastic Kubernetes Service (EKS)

O Fargate facilita a sua concentração no desenvolvimento de aplicativos. O Fargate elimina a necessidade de provisionar e gerenciar servidores, permite que você especifique e pague pelos recursos por aplicativo, além de aumentar a segurança ao conceber aplicativos isolados

O Fargate aloca a quantidade certa de computação, eliminando a necessidade de escolher instâncias e ajustar a escala da capacidade do cluster. Você só paga pelos recursos exigidos para a execução dos contêineres, por isso não há excesso de provisionamento nem pagamento por servidores adicionais

O Fargate executa cada tarefa ou pod no próprio kernel do serviço, disponibilizando às tarefas e aos pods ambientes próprios isolados de computação. Isso permite que o aplicativo seja concebido para oferecer isolamento da carga de trabalho e segurança otimizada

Como Funciona



The Fargate launch type allows you to run your containerized applications without the need to provision and manage the backend infrastructure. Just register your task definition and Fargate launches the container for you.

Links Úteis

- https://docs.aws.amazon.com/AmazonECS/latest/developerguide/launch_types.html
- https://docs.aws.amazon.com/AmazonECS/latest/developerguide/AWS_Fargate.html

AWS Lambda

AWS Lambda is a compute service that lets you run code without provisioning or managing servers. AWS Lambda executes your code only when needed and scales automatically, from a few requests per day to thousands per second. You pay only for the compute time you consume - there is no charge when your code is not running. With AWS Lambda, you can run code virtually for any type of application or backend service - all with zero administration.

O AWS Lambda permite que se execute código sem provisionar ou gerenciar servidores. Paga-se apenas pelo tempo de computação consumido

Basta carregar o código e o Lambda se encarrega de todos os itens necessários para executar e permitir que o código seja escalável e com alta disponibilidade

O AWS Lambda executa automaticamente o código sem exigir que você provisione ou gerencie servidores. Basta escrever o código e fazer upload para o Lambda

A Escalabilidade é sempre. O código é executado em paralelo e processa cada acionamento individualmente, escalando precisamente continua. O Lambda escala automaticamente as aplicações executando código em resposta a cada acionamento de acordo com o tamanho da carga de trabalho, ou seja, se chegar 5 requisições Lambdas, cada uma será processada individualmente em paralelo

Possui um **medidor de fração de segundo**. Com o AWS Lambda, você é **cobrado a cada 100ms de execução** de código **e pelo número de vezes que o código é acionado**. Pagando apenas pelo tempo de computação consumido

Performance uniforme. Com o AWS Lambda, é possível otimizar o tempo de execução do código escolhendo o tamanho de memória ideal para a função

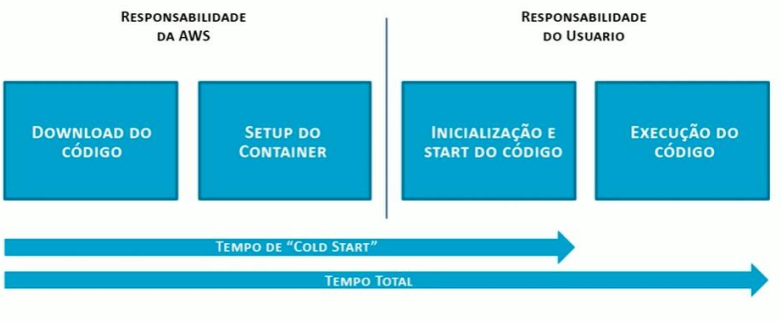
É possível habilitar a **Simultaneidade Provisionada** para manter as funções inicializadas e prontas para responder em questão de poucos milissegundos. O processamento da função acaba sendo um pouco mais caro porém, a cobrança não é realizada pelo tempo todo da função ativa pois se não, não faria sentido

Start do Lambda:

Quando o Lambda é acionado, acontece o download do código, a uma altíssima velocidade, e faz o setup do container, ou seja, a AWS está montando o ambiente que o código vai ser executado

A execução do código não entra no tempo de “Cold Start”, que é o tempo de aquecer o lambda

Quanto mais conciso for o código e mais rápido ele ser executado, menor será o tempo de execução



Source : AWS re:Invent 2017 – Become a serverless Black Belt

Exemplos:

A cobrança é realizada pelo número de solicitações de suas funções e pela duração, o tempo que leva para que seu código seja executado

O Lambda conta uma solicitação cada vez que começa a executar em resposta a uma notificação de evento ou chamada de invocação, incluindo invocações de teste do console

A duração é calculada a partir do momento em que seu código começa a ser executado até ele retornar ou encerrar, arredondando para os **1 ms** mais próximos

O preço depende da quantidade de memória que você alocar para sua função. A capacidade de CPU e outros recursos são alocados de forma proporcional

Um aumento no tamanho da memória aciona um aumento equivalente na CPU disponível para a função

Todas as chamadas feitas para o AWS Lambda devem completar a execução dentro de **15 min**. **O timeout default é de 15 segundos**, mas é possível setar o timeout para qualquer valor entre 1 segundo e 15 min

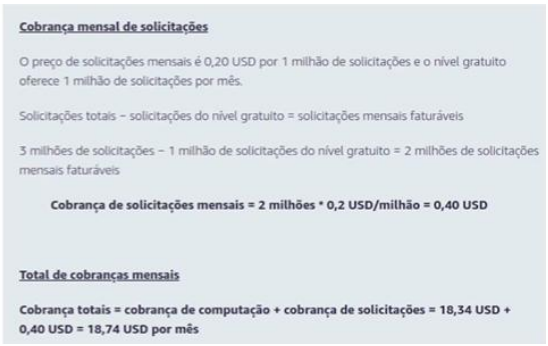
O nível de uso gratuito da AWS Lambda inclui 1 milhão de solicitações gratuitas e 400.000 GB/segundos de tempo de computação por mês (400.000 segundos utilizando 1 GB de memória por mês)

Região: Leste dos EUA (Norte da Virgínia)

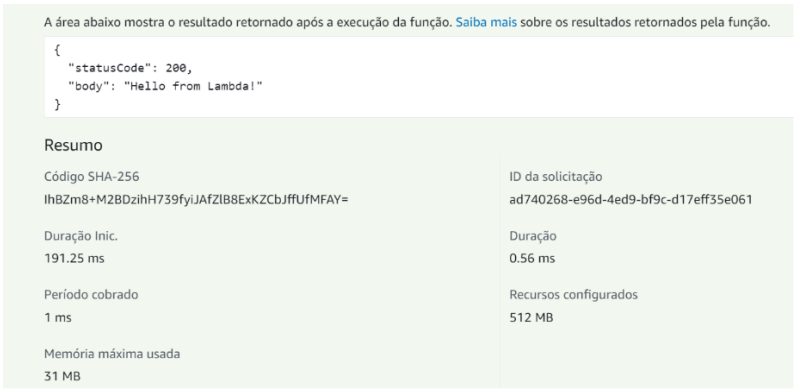
	Preço
Solicitações	0,20 USD por 1 milhão de solicitações
Duração	0,0000166667 USD por cada GB/segundo

Memória (MB)	Preço por 100 ms (USD)
128	0,0000002083 USD
512	0,0000008333 USD
1.024	0,0000016667 USD
1.536	0,0000025000 USD
2.048	0,0000033333 USD
3.008	0,0000048958 USD

Exemplo de Cobrança:



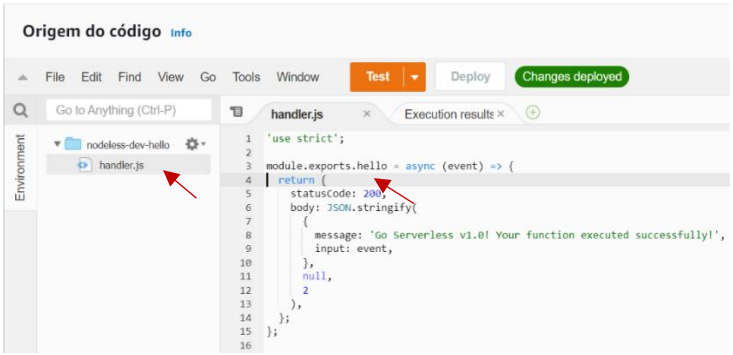
Testando uma Função Lambda:

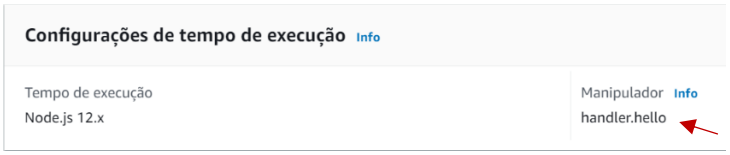


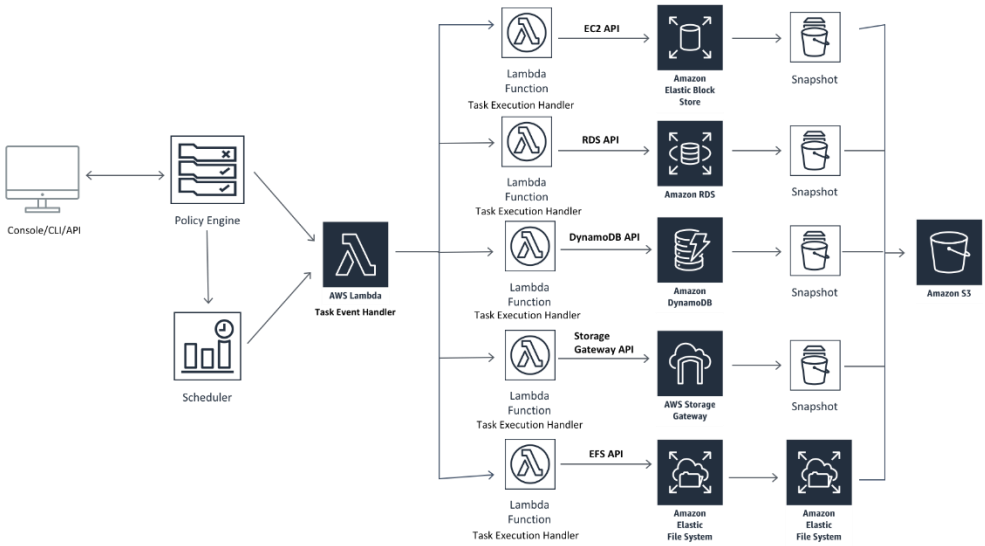
A primeira execução do teste mostra o campo “Duração Inic.” que é apresentada toda vez que uma função é executada pela primeira vez, ou quando ela passa um tempo sem ser executada, caso uma nova requisição ocorra em seguida esse campo não será apresentado pois a função ainda está ativa em background, ficando inativa depois de alguns segundos

O campo “Memória máxima usada” de 31 MB demonstra que o recurso configurado de 512 MB pode ser baixado para 128 MB

O nome do manipulador deve ser constituído pelo nome do arquivo + o nome da function:







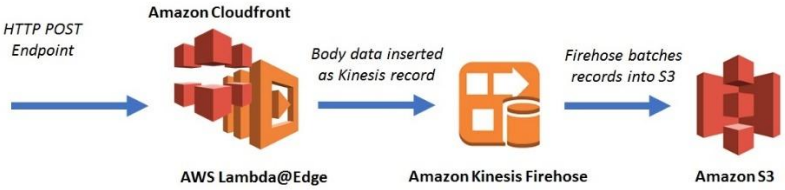
CloudFront and Lambda@Edge

O Lambda @ Edge permite que você execute funções do AWS Lambda globalmente para que possa processar e responder às solicitações do usuário em latências baixas. A execução de funções Lambda em estreita proximidade geográfica com os usuários ajuda a satisfazer uma série de casos de uso, como personalização de site, Search Engine Optimization (SEO), reescrita de URL e teste A / B - apenas para citar alguns. Hoje anunciamos que Lambda @ Edge agora pode acessar o HTTP Request Body.

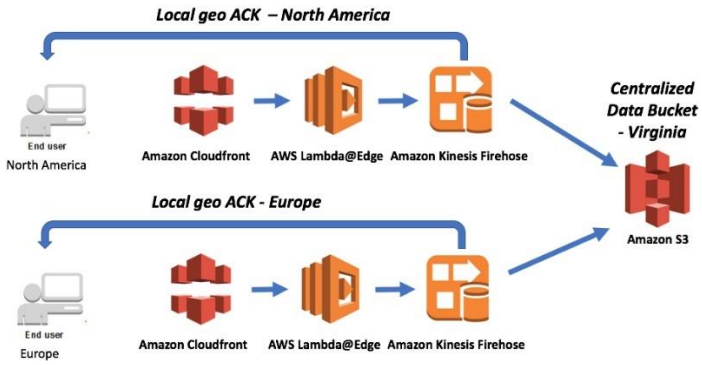
Essa funcionalidade permite que novos casos de uso aproveitem os benefícios da computação de ponta. Nesta postagem, daremos uma olhada em um cenário específico: uma passagem de ingestão global de dados por meio do Amazon CloudFront, Lambda @ Edge e Amazon Kinesis Firehose para o Amazon S3. Você pode usar o Amazon Kinesis Firehose como um mecanismo de ingestão de streaming sem servidor para muitos tipos diferentes de dados, desde arquivos de log a dados de impressão.

Ao ingerir dados dos produtores, os dados podem ser armazenados de forma duradoura e estão prontamente disponíveis para análise. Muitos clientes usam os SDKs da AWS ou a Biblioteca do Produtor Kinesis para ingerir dados ou instalam Agentes Kinesis para ingestão e facilidade de uso. Mas quando seus produtores de dados são distribuídos por um número altamente volátil de clientes, você precisa de um serviço escalonável para lidar com o tráfego. Um exemplo disso é quando os clientes apenas usam navegadores da web, e incorporar a biblioteca SDK / produtor em seu site pode não ser fácil.

Como o Amazon Cloudfront fala HTTP simples (sem AWS SDK necessário), os javascripts do lado do cliente (exemplo: web bugs / beacons etc) podem se comunicar facilmente com o endpoint HTTP do CloudFront. Com a integração do CloudFront com Lambda @ Edge, você pode criar uma camada de ingestão com o Amazon Kinesis Firehose usando apenas algumas etapas de configuração simples e linhas de código. Depois que os dados são inseridos no Kinesis Firehose, eles podem ser salvos de forma duradoura em uma solução de armazenamento como o Amazon S3. Embora estejamos usando o Amazon Kinesis Firehose como exemplo neste blog, o Amazon Kinesis Streams também funciona. A seguir estão os serviços que usaremos:



Para ajudar essa solução a se expandir globalmente, os usuários em diferentes localidades receberão um reconhecimento geográfico local de que a mensagem foi recebida. Responder localmente é muito mais rápido do que enviar primeiro tráfego para a origem, conforme mostrado no diagrama a seguir:



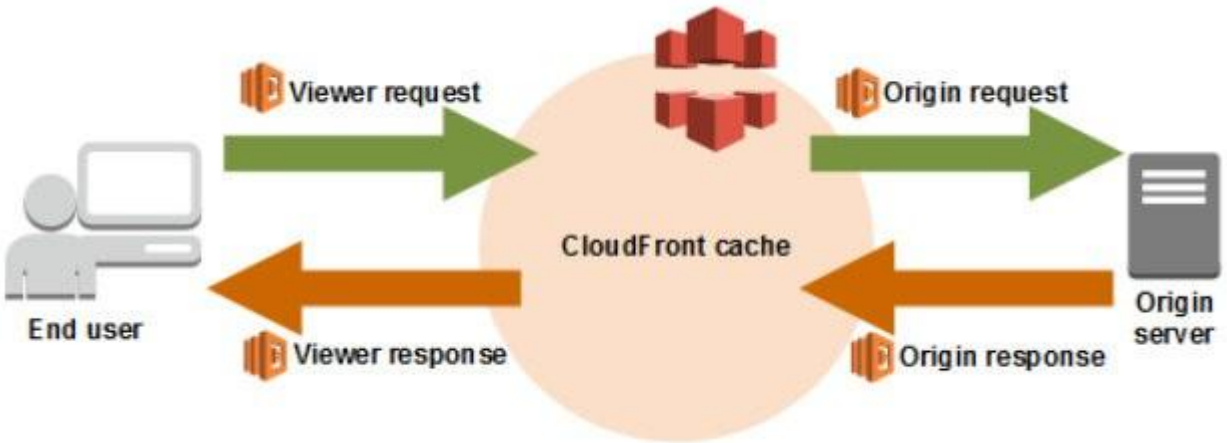
Outra vantagem de usar o CloudFront junto com Lambda @ Edge é que você pode usar a segurança integrada que o AWS Web Application Firewall (WAF) oferece. Ao usar o WAF, você pode colocar endereços IP de clientes na lista de permissões e na lista negra e também está protegido contra o tráfego de outros aplicativos maliciosos.

Events that can trigger AWS Lambda@edge function

Q: What Amazon CloudFront events can be used to trigger my functions?

Your functions will automatically trigger in response to the following Amazon CloudFront events:

- Viewer Request - This event occurs when an end user or a device on the Internet makes an HTTP(S) request to CloudFront, and the request arrives at the edge location closest to that user.
- Viewer Response - This event occurs when the CloudFront server at the edge is ready to respond to the end user or the device that made the request.
- Origin Request - This event occurs when the CloudFront edge server does not already have the requested object in its cache, and the viewer request is ready to be sent to your backend origin webserver (e.g. Amazon EC2, or Application Load Balancer, or Amazon S3).
- Origin Response - This event occurs when the CloudFront server at the edge receives a response from your backend origin webserver.



Lambda Acessando a Internet e Recursos Internos de uma VPC

As funções do AWS Lambda podem ser executadas em um VPC privado com os recursos alocados dentro da sub-rede fornecida durante a configuração.

Para acessar os recursos privados da Amazon VPC, como uma instância de banco de dados do Relational Database Service (Amazon RDS) ou uma instância do Amazon Elastic Compute Cloud (Amazon EC2), associe a função do Lambda de uma Amazon VPC a uma ou mais sub-redes privadas.

Para conceder à função acesso à Internet, a VPC associada precisa ter um gateway NAT (ou uma instância NAT) em uma sub-rede pública.

Observação: a caraterística da sub-rede, privada ou pública, depende da tabela de rotas. As sub-redes públicas têm uma rota que indica um gateway da Internet, e as privadas não.

Todas as funções do Lambda são executadas com segurança dentro de uma nuvem privada virtual (VPC) gerenciada pelo sistema padrão. No entanto, você também pode configurar sua função Lambda para acessar recursos em um VPC personalizado. Um VPC personalizado define uma rede privada de recursos, como bancos de dados, instâncias de cache ou serviços internos.

Se as permissões do AWS Identity and Access Management (IAM) permitem que você apenas crie funções que se conectam ao seu VPC, você deve configurar os detalhes do VPC ao criar a função. Se suas permissões de IAM permitem que você crie funções que não estão conectadas ao seu VPC, você pode adicionar a configuração do VPC depois de criar a função.

Observação: Para que a função lambda acesse o endpoint de serviço S3 de dentro do VPC privado, deve haver um gateway NAT ou um endpoint S3 VPC configurado na tabela de rota associada à sub-rede que foi escolhida durante a configuração da função Lambda. Caso contrário, o tempo limite da solicitação se esgotaria.

Erros Comuns

If the function reaches the maximum configured memory, in this case
128 MB, the function gets terminated with an error message as below, not as **request timed out**.
REPORT RequestId: xxxxxxxx Duration: xxxxx ms Billed Duration: xxxxx ms
Memory Size: 128 MB Max Memory Used: 129 MB RequestId: xxxxxxx Process exited before completing request

Lambda Ephemeral Disk

AWS Lambda Limits

AWS Lambda Resource Limits per Invocation

Resource	Limits
Memory allocation range	Minimum = 128 MB / Maximum = 3008 MB (with 64 MB increments). If the maximum memory use is exceeded, function invocation will be terminated.
Ephemeral disk capacity ("/tmp" space)	512 MB

Invoke Synchronously and Asynchronously

The following are the functions that invoke synchronously and asynchronously the AWS Lambda function.

Services That Invoke Lambda Functions Synchronously

- Elastic Load Balancing (Application Load Balancer) (p. 170)
- Amazon Cognito (p. 216)
- Amazon Lex (p. 265)
- Amazon Alexa (p. 172)
- Amazon API Gateway (p. 173)
- Amazon CloudFront (Lambda@Edge) (p. 205)
- Amazon Kinesis Data Firehose (p. 264)
- AWS Step Functions
- Amazon Simple Storage Service Batch (p. 285)

For asynchronous invocation, Lambda queues the event before passing it to your function. The other service gets a success response as soon as the event is queued and isn't aware of what happens afterwards. If an error occurs, Lambda handles retries (p. 117), and can send failed events to a dead-letter queue (p. 104) that you configure.

Services That Invoke Lambda Functions Asynchronously

- Amazon Simple Storage Service (p. 270)
- Amazon Simple Notification Service (p. 289)
- Amazon Simple Email Service (p. 287)
- AWS CloudFormation (p. 203)
- Amazon CloudWatch Logs (p. 202)
- Amazon CloudWatch Events (p. 196)
- AWS CodeCommit (p. 207)
- AWS Config (p. 217)
- AWS IoT (p. 246)
- AWS IoT Events (p. 247)
- AWS CodePipeline (p. 208)

Casos de Uso

- Periodically check the log files for errors in CloudWatch or CloudTrail and send out notifications through SNS.
- Download S3 bucket objects of size varying between 1 MB - 512 MB to a Lambda Ephemeral disk or temp location, read and analyze them for keywords and add the keywords to the metadata of file object for search purposes.
- Schedule a job and invoke a Lambda function to generate AWS resource usage reports based on certain tags.
- A website with highly scalable backend layer that will persist data into RDS or DynamoDB.
- You can host the web frontend on S3 and accelerate content delivery with Cloudfront caching. The web frontend can send requests to Lambda functions via API Gateway HTTPS endpoints. Lambda can handle the application logic and persist data to a fully managed database service.
- (RDS for relational, or DynamoDB for non-relational database). You can host your Lambda functions and databases within a VPC to isolate them from other networks.

Function Configuration, deployments and execution

The following quotas apply to function configuration, deployments, and execution. They cannot be changed.

Resource	Quota
Function memory allocation	128 MB to 10,240 MB, in 1-MB increments.
Function timeout	900 seconds (15 minutes)
Function environment variables	4 KB
Function resource-based policy	20 KB
Function layers	5 layers
Function burst concurrency	500 - 3000 (varies per Region)
Invocation payload (request and response)	6 MB (synchronous) 256 KB (asynchronous)
Deployment package (.zip file archive) size	50 MB (zipped, for direct upload) 250 MB (unzipped, including layers)

Minimum subnet range

You have a requirement to create an AWS Lambda function inside a private VPC which will be communicating with the RDS instance inside the same private VPC. You have set up the memory to be 1 GB for the Lambda function. You expect concurrent requests during peak to be 100 per sec, and the average Function execution time is 1 sec. What should be the minimum subnet range you must choose to create a subnet to run the Lambda function without any issues successfully?

- If your Lambda function accesses a VPC, you must make sure that your VPC has sufficient ENI capacity to support the scale requirements of your Lambda function. You can use the following formula to approximately determine the ENI capacity.

```
Projected peak concurrent executions * (Memory in GB / 3GB)
```

Where:

  - **Projected peak concurrent execution** – Use the information in Managing Concurrency to determine this value.
  - **Memory** – The amount of memory you configured for your Lambda function.

Peak concurrent executions = 100 * 1 = 100
ENI Capacity = 100 * (1GB / 3GB) = 33.33 i.e. = 33
Hence we need /26 CIDR

- /24 CIDR range comes with 256 IP address with 251 available IP addresses
- /25 CIDR range comes with 128 IP address with 123 available IP addresses.
- /26 CIDR range comes with 64 IP addresses with 59 available IP addresses.
- /27 CIDR range comes with 32 IP addresses with 27 available IP addresses.

Using AWS Lambda environment variables

You can use environment variables to adjust your function's behavior without updating code. An environment variable is a pair of strings that is stored in a function's version-specific configuration. The Lambda runtime makes

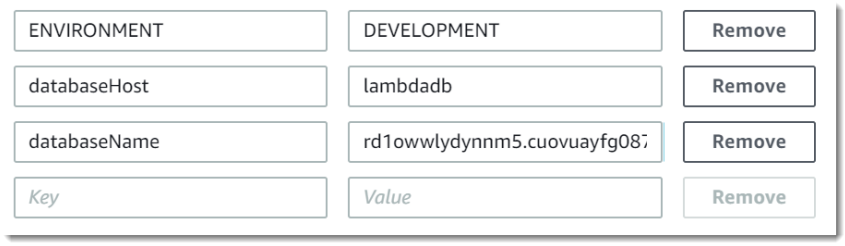
environment variables available to your code and sets additional environment variables that contain information about the function and invocation request.

To increase database security, we recommend that you use **AWS Secrets Manager** instead of environment variables to store database credentials.

Example scenario for environment variables

You can use environment variables to customize function behavior in your test environment and production environment. For example, you can create two functions with the same code but different configurations. One function connects to a test database, and the other connects to a production database. In this situation, you use environment variables to tell the function the hostname and other connection details for the database.

The following example shows how to define the database host and database name as environment variables.



Retrieve environment variables

To retrieve environment variables in your function code, use the standard method for your programming language.

- `getFunctionVersion()`: The Lambda function version that is executing. If an alias is used to invoke the function, then `getFunctionVersion` will be the version the alias points to.

AWS_LAMBDA_FUNCTION_VERSION	Yes	The version of the Lambda function.

Lambda function aliases

You can create one or more aliases for your Lambda function. A Lambda alias is like a pointer to a specific function version. Users can access the function version using the alias Amazon Resource Name (ARN).

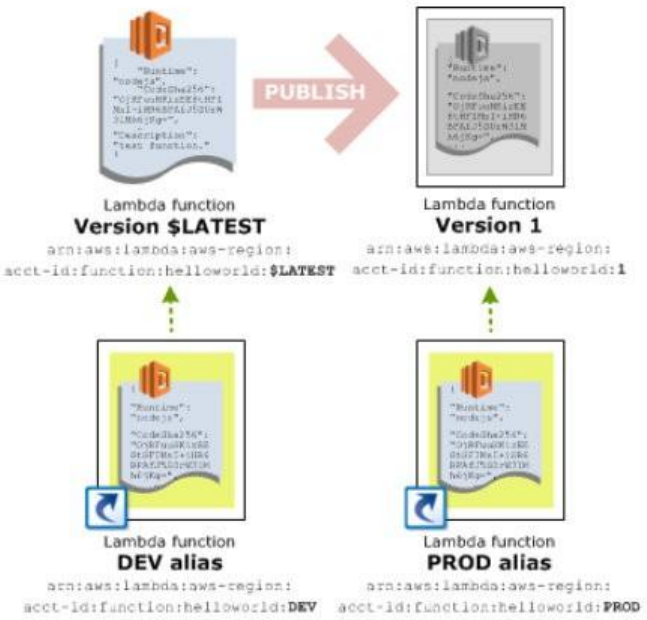
AWS Lambda aliases enable the following use cases:

- **Easier support for promotion of new versions of Lambda functions and rollback when needed** – After initially creating a Lambda function (the `$LATEST` version), you can publish a version 1 of it. By creating an alias named PROD that points to version 1, you can now use the PROD alias to invoke version 1 of the Lambda function.
  Now, you can update the code (the `$LATEST` version) with all of your improvements, and then publish another stable and improved version (version 2). You can promote version 2 to production by remapping the PROD alias so that it points to version 2. If you find something wrong, you can easily roll back the production version to version 1 by remapping the PROD alias so that it points to version 1.

  Note

  In this context, the terms *promotion* and *roll back* refer to the remapping of aliases to different function versions.

- **Simplify management of event source mappings** – Instead of using Amazon Resource Names (ARNs) for Lambda function in event source mappings, you can use an alias ARN. This approach means that you don't need to update your event source mappings when you promote a new version or roll back to a previous version.

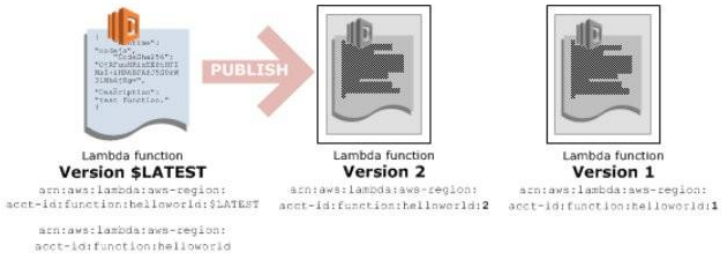


Publishing an AWS Lambda Function Version

When you publish a version, AWS Lambda makes a snapshot copy of the Lambda function code (and configuration) in the $LATEST version. A published version is immutable. That is, you can't change the code or configuration information. The new version has a unique ARN that includes a version number suffix as shown following.

You can publish multiple versions of a Lambda function. Each time you publish a version, AWS Lambda copies $LATEST version (code and configuration information) to create a new version. When you publish additional versions, AWS Lambda assigns a monotonically increasing sequence number for versioning, even if the function was deleted and recreated. Version numbers are never reused, even for a function that has been deleted and recreated. This approach means that the consumer of a function version can depend on the executable of that version to never change (except if it's deleted).

If you want to reuse a qualifier, use aliases with your versions. Aliases can be deleted and re-created with the same name.



You can refer to this function using its Amazon Resource Name (ARN). There are two ARNs associated with this initial version:

- **Qualified ARN** – The function ARN with the version suffix.

```
arn:aws:lambda:aws-region:acct-id:function:helloworld:$LATEST
```

- **Unqualified ARN** – The function ARN without the version suffix.
  You can use this unqualified ARN in all relevant operations. However, you cannot use it to create an alias. For more information, see Introduction to AWS Lambda Aliases.
  The unqualified ARN has its own resource policies.

```
arn:aws:lambda:aws-region:acct-id:function:helloworld
```

Note

Unless you choose to publish versions, the $LATEST function version is the only Lambda function version that you have. You can use either the qualified or unqualified ARN in your event source mapping to invoke the $LATEST version.

You can access the function using either the function ARN or the alias ARN.

- Because the function version for an unqualified function always maps to $LATEST, you can access it using the qualified or unqualified function ARN. The following shows a qualified function ARN with the $LATEST version suffix.

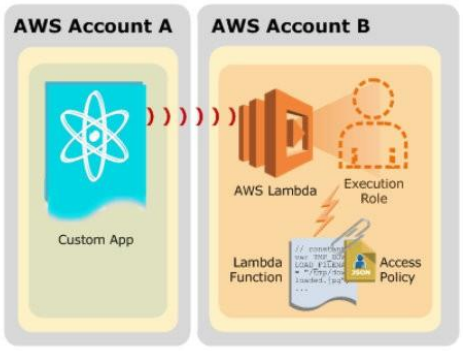
```
arn:aws:lambda:aws-region:acct-id:function:helloworld:$LATEST
```

- When using any of the alias ARNs, you are using a qualified ARN. Each alias ARN has an alias name suffix.

```
arn:aws:lambda:aws-region:acct-id:function:helloworld:PROD
arn:aws:lambda:aws-region:acct-id:function:helloworld:BETA
arn:aws:lambda:aws-region:acct-id:function:helloworld:DEV
```

Grant cross-account permissions

You can also grant cross-account permissions using the function policy. For example, if a user-defined application and the Lambda function it invokes belong to the same AWS account, you don't need to grant explicit permissions. Otherwise, the AWS account that owns the Lambda function must allow cross-account permissions in the permissions policy associated with the Lambda function.



Example 3: Allow a User Application Created by Another AWS Account to Invoke a Lambda Function (Cross-Account Scenario)

To grant permissions to another AWS account (that is, to create a cross-account scenario), you specify the AWS account ID as the `principal` value as shown in the following AWS CLI command:

```
aws lambda add-permission \
--region region \
--function-name helloworld \
--statement-id 3 \
--principal 111111111111 \
--action lambda:InvokeFunction \
--profile adminuser
```

In response, AWS Lambda returns the following JSON code. The `Statement` value is a JSON string version of the statement added to the Lambda function policy.

```
{
    "Statement": "{\"Action\":[\"lambda:InvokeFunction\"],
                  \"Resource\":\"arn:aws:lambda:us-west-2:account-id:function:helloworld\",
                  \"Effect\":\"Allow\",
                  \"Principal\":{\"AWS\":\"account-id\"},
                  \"Sid\":\"3\"}"
}
```

Links Úteis

https://aws.amazon.com/pt/blogs/networking-and-content-delivery/global-data-ingestion-with-amazon-cloudfront-and-lambdaedge/

https://aws.amazon.com/pt/premiumsupport/knowledge-center/internet-access-lambda-function/

https://aws.amazon.com/about-aws/whats-new/2018/10/aws-lambda-supports-functions-that-can-run-up-to-15-minutes/

https://aws.amazon.com/blogs/compute/robust-serverless-application-design-with-aws-lambda-dlq/

https://docs.aws.amazon.com/lambda/latest/dg/with-scheduled-events.html?shortFooter=true

https://docs.aws.amazon.com/lambda/latest/dg/configuration-envvars.html

➢ Pontos de Atenção

1. You are working as a Solutions Architect for a leading data analytics company in which you are tasked to process real-time streaming data of your users across the globe. This will enable you to track and analyze globally-distributed user activity on your website and mobile applications, including click stream analysis. Your cloud architecture should process the data in close geographical proximity to your users and to respond to user requests at low latencies. Which of the following options is the most ideal solution that you should implement? R: Integrate CloudFront with Lambda@Edge in order to process the data in close geographical proximity to users and respond to user requests at low latencies. Process real-time streaming data using Kinesis and durably store the results to an Amazon S3 bucket.

2. You are uploading large files to AWS S3 bucket, ranging from 1GB – 3GB. Your organization has a requirement to calculate the hash checksum of the file by reading entire file so the users can validate the checksum to identify any potential corruptions during downloads. For this, you created a Lambda function and getting it triggered through S3 notifications. However, the request is getting timed out. What could be the reason? R: Lambda function is set to run in a private VPC without NAT Gateway or VPC Endpoint.

3. Your organization must perform big data analysis to transform data and store the result in the AWS S3 bucket. They have implemented the solution using AWS Lambda due to its zero-administrative maintenance and cost-effective nature. However, in very few cases, the execution is getting abruptly terminated after 15 minutes. They would like to get a notification in such scenarios. What would you do? R: Configure Dead-letter Queue and send a notification to SNS topic. You can forward non-processed payloads to Dead Letter Queue (DLQ) using AWS SQS, AWS SNS.

4. Your organization uploads relatively large compressed files ranging between 100MB – 200MB in size to AWS S3 b ucket. Once uploaded, they are looking to calculate the total number objects in the compressed file and add the total count as a metadata to the compressed file in AWS S3. They approached you for a cost-effective solution. You have recommended using AWS Lambda through S3 event notifications to perform this op eration. However, they were concerned about failures as S3 event notification is an asynchronous one-time trigger and Lambda can fail due to operation time outs, max memory limits, max execution time limits etc. What is the best retry approach you recommend? R: Configure Dead-letter queue with SQS. Configure SQS to trigger Lambda function again